NAVAL POSTGRADUATE SCHOOL

MONTEREY, CALIFORNIA

THESIS

MACHINE LEARNING TECHNIQUES FOR PERSUASION DECTECTION IN CONVERSATION

by

Pedro Ortiz

June 2010

Thesis Advisor:	Craig H. Martell
Second Reader:	Joel D. Young

Approved for public release; distribution is unlimited

THIS PAGE INTENTIONALLY LEFT BLANK

REPORT DOCUMENTATION PAGE

Form Approved OMB No. 0704–0188

1. REPORT DATE *(DD–MM–YYYY)*	2. REPORT TYPE	3. DATES COVERED *(From — To)*
17–6–2010	Master's Thesis	2008-06-01—2010-06-31

4. TITLE AND SUBTITLE

Machine Learning Techniques for Persuasion Dectection in Conversation

5a. CONTRACT NUMBER

5b. GRANT NUMBER

5c. PROGRAM ELEMENT NUMBER

6. AUTHOR(S)

Pedro Ortiz

5d. PROJECT NUMBER

5e. TASK NUMBER

5f. WORK UNIT NUMBER

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)

Naval Postgraduate School
Monterey, CA 93943

8. PERFORMING ORGANIZATION REPORT NUMBER

9. SPONSORING / MONITORING AGENCY NAME(S) AND ADDRESS(ES)

Department of the Navy

10. SPONSOR/MONITOR'S ACRONYM(S)

11. SPONSOR/MONITOR'S REPORT NUMBER(S)

12. DISTRIBUTION / AVAILABILITY STATEMENT

Approved for public release; distribution is unlimited

13. SUPPLEMENTARY NOTES

The views expressed in this thesis are those of the author and do not reflect the official policy or position of the Department of Defense or the U.S. Government. **IRB Protocol Number: n/a**

14. ABSTRACT

We determined that it is possible to automatically detect persuasion in conversations using three traditional machine learning techniques, naive bayes, maximum entropy, and support vector machine. These results are the first of their kind and serve as a baseline for all future work in this field. The three techniques consistently outperformed the baseline F-score, but not at a level that would be useful for real world applications. The corpus of data was comprised of four types of negotiation transcripts, labeled according to a persuasion model developed by James Cialdini. We discovered that the transcripts from the Davidian standoff in Waco, Texas were significantly different from the rest of the corpus. We have included suggestions for future work in the areas of data set improvements, feature set improvements, and additional research. Advancements in this field will contribute to the Global War on Terror by alerting intelligence analysts to enemy persuasion attempts and by enabling U.S. forces to conduct more effective information and psychological operations using local persuasion models.

15. SUBJECT TERMS

Persuasion, Conversation, Machine Learning, Naive Bayes, Maximum Entropy, Support Vector Machine, Waco, David Koresh, Negotiation, Transcript, Cialdini

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT	b. ABSTRACT	c. THIS PAGE			19b. TELEPHONE NUMBER *(include area code)*
Unclassified	Unclassified	Unclassified	UU	131	

Standard Form 298 (Rev. 8–98)
Prescribed by ANSI Std. Z39.18

THIS PAGE INTENTIONALLY LEFT BLANK

Approved for public release; distribution is unlimited

MACHINE LEARNING TECHNIQUES FOR PERSUASION DECTECTION IN CONVERSATION

Pedro Ortiz
Captain, United States Marine Corps
B.S., University of Pennsylvania, 2004

Submitted in partial fulfillment of the
requirements for the degree of

MASTER OF SCIENCE IN COMPUTER SCIENCE

from the

NAVAL POSTGRADUATE SCHOOL
June 2010

Author: Pedro Ortiz

Approved by: Craig H. Martell
Thesis Advisor

Joel D. Young
Second Reader

Peter J. Denning
Chair, Department of Computer Science

THIS PAGE INTENTIONALLY LEFT BLANK

ABSTRACT

We determined that it is possible to automatically detect persuasion in conversations using three traditional machine learning techniques, naive bayes, maximum entropy, and support vector machine. These results are the first of their kind and serve as a baseline for all future work in this field. The three techniques consistently outperformed the baseline F-score, but not at a level that would be useful for real world applications. The corpus of data was comprised of four types of negotiation transcripts, labeled according to a persuasion model developed by James Cialdini. We discovered that the transcripts from the Davidian standoff in Waco, Texas were significantly different from the rest of the corpus. We have included suggestions for future work in the areas of data set improvements, feature set improvements, and additional research. Advancements in this field will contribute to the Global War on Terror by alerting intelligence analysts to enemy persuasion attempts and by enabling U.S. forces to conduct more effective information and psychological operations using local persuasion models.

THIS PAGE INTENTIONALLY LEFT BLANK

Table of Contents

THIS PAGE INTENTIONALLY LEFT BLANK

List of Figures

THIS PAGE INTENTIONALLY LEFT BLANK

List of Tables

THIS PAGE INTENTIONALLY LEFT BLANK

Acknowledgements

I would like to acknowledge the many people who made this work possible. I would first like to thank my thesis advisor, Professor Craig Martell. You "primed my intuition pump", helping me to become a computer scientist. I would like to thank Lieutenant Colonel Joel Young, USAF. Your critical eye and valuable insight helped to make this thesis "outstanding." Without you, my writing would never have been as good, and without Professor Martell, it would have never been finished. I would like to thank Dr. Andrew Schein for his hours of system administration, theory explanation, and side-by-side coding. You showed me that coding can be a spectator sport. I would like to thank Professor Dennis Volpano. Both of your courses, Automata 1 and Automata 2, challenged and prepared me for the work that lay ahead. I hope that some day people will say that I am as good a Marine as the four of these men are computer scientists.

I would like to thank the Marines and Sailors in my cohort without whom I would have never learned so much. I would like to thank Capt Justin Jones, USMC and Lieutenant Dennis Holden, USN for their help and friendship as we trudged along the academic path. I wish you fair winds and following seas.

I would like to thank my parents, Hector and Raquel Ortiz, for their support and guidance without which I would have never received the education I have today. Lastly and most importantly, I would like to thank my wife, Elizabeth Ortiz. Without your support, patience, and understanding, this work would never have been completed. I would have died in the lab of either exhaustion or starvation.

THIS PAGE INTENTIONALLY LEFT BLANK

CHAPTER 1:
Introduction

Nearly every person has an item in their house that is sitting in their closet collecting dust. This item has only ever been outside the closet on the day it was brought home. For every person, it is a different item. For some, it is a set of ugly dishes given to them by their grandmother, for others a lifetime supply of the latest and greatest cleaning product on the infomercial circuit. Regardless of the item, it was never really wanted in the first place, yet there it sits. How did this happen? The answer is persuasion. Persuasion can do more than just get Americans to fill their closets. It can convince people to plant bombs on Times Square or to crash a plane into the Pentagon. Persuasion can take on many forms and those with knowledge and awareness of these forms can sell a ketchup popsicle to a woman wearing white gloves, or even save the lives of thousands of Americans.

1.1 Motivation

The motivation for this research stems from the following ideas expressed in *Joint Vision 2020* [1].

> Information, information processing, and communications networks are at the core of every military activity.
>
> Information superiority provides the joint force a competitive advantage only when it is effectively translated into *superior knowledge* and decisions.

If it were possible to detect persuasion in conversation, troops on the front lines of our current and future wars would have more information to contribute to their knowledge base and from which to base decisions. They would have invaluable information, such as notifications about enemies trying to influence the local populace, the target audience of these persuasion attempts, and what persuasion model the enemy is using. Information of this type would result in better intelligence targeting, more focused information operations, and application of localized persuasion models to influence the local populace in support of U.S. interests.

1.2 Related Work

There exists previous work in related areas but very little work on actual persuasion detection. One group of researchers investigated if it is possible to automatically determine from which perspective an essay was written [2]. This research used a corpus of data from the "bitter-lemons" Web site, and addressed the binary classification of perspectives between Israeli and Palestinian authors. Another group of researchers examined the problem of sentiment detection [3]. They used consumer reviews in order to determine whether or not it is possible to discriminate between different star ratings for a given review. The only work directly focused on persuasion is a product of the Naval Postgraduate School [4]. Research was conducted to evaluate the degree to which a persuasion model could be used in order to annotate a corpus of data for machine learning experiments. This research used a persuasion model developed by James Cialdini and resulted in the only known corpus of persuasion tagged data in existence.

1.3 Research Question

This thesis addresses the question, "Can we learn to identify persuasion as characterized by Cialdini's model using traditional machine learning techniques?" This research uses four different feature sets and three different machine learning techniques in order to answer this question. Additionally, this research explores the role of feature discrimination, types of segmentation, and voting schemes.

1.4 Results

The results of this research allow us to answer our research question with a "yes." However, none of the methods used, neither separately nor combined, produced the type of results that would allow us to consider this problem solved. This research produced some weak classifiers and identified some candidate feature sets for future research in this field. One type of segmentation failed to produce any learnable signal. The results of the weak classifiers were used in conjunction with two different voting schemes that yielded mixed results.

1.5 Future Work

Future work in this area falls into three categories: data set improvements, feature set improvements, and future research. Data set improvements could include producing more and larger data sets annotated for belief, adding additional genres such as Web pages, blogs, and SMS messages, and augmenting the current data set with additional information, such as distance

from the previous persuasive post, correct speaker tags and dialogue act tags. Features set improvements should focus on combining high recall features with high precision features and building topic models for persuasion. In order to solve this problem, future research is needed to investigate segmentation schemes, effects of time and sequence, the utility of bagging, boosting, and voting, the role of speaker type, and the impact of parts of speech and syntax.

1.6 Organization of Thesis

In order to investigate the research question, this thesis is organized as follows:

- Chapter 1 discusses the topic of persuasion and the motivation for techniques to automatically detect persuasion in conversation.
- Chapter 2 discusses prior work relevant to the task of persuasion detection.
- Chapter 3 contains a description of the experimental design and the data set used in this research.
- Chapter 4 contains the results of the experiments and analysis of the results.
- Chapter 5 contains concluding remarks and possible areas of future research.

THIS PAGE INTENTIONALLY LEFT BLANK

CHAPTER 2:
Prior and Related Work

2.1 Introduction

Learning to automatically identify persuasion requires us to define and understand what is meant by "persuasion." In this chapter, we present all concepts relevant to detecting persuasion in conversation using machine learning techniques. First, we present the persuasion model used in this research. From this foundation, we survey feature definitions, followed by an overview of three machine learning classification techniques. Next, we discuss a suite of metrics needed to evaluate our hypotheses. We conclude with a discussion of the software tools that enabled this research.

2.2 Persuasion

Persuasion is an everyday social phenomenon. For persuasion to be present, one party must be unwilling or unlikely to perform an act or to believe an idea unless they are influenced by an outside force. This force can manifest in another person, an advertisement, or current social norms and practices. One formal definition of persuasion by James Cialdini identifies six types of persuasion: *reciprocity, commitment and consistency, liking, authority, social proof,* and *scarcity* [5]. Since Cialdini's model forms the foundation for this research, it is important to have an understanding of what characterizes each type of persuasion.

2.2.1 Reciprocity

Reciprocity relies on a condition of indebtedness. According to the Cialdini persuasion model, this behavior developed out of a need for society to advance. When two cultures meet, they exchange ideas. The mechanism of *reciprocity* allows that exchange to continue. One well known example of this principle is the Hari Krishna monks. These monks offer flowers to people in public places. Despite the fact that this is a small gift, people will in turn give a dollar or two back to the monks. Often, people will throw the flowers away and the monks simply retrieve them from the garbage and repeat the process [5].

Reciprocity can take on another form in which the exchange is not a tangible item; instead the exchange is an exchange of utility. The following is an example of this principle at work.

> A Boy Scout walks up to a man and asks him to buy a few tickets at a cost of five dollars for their two hour talent show this coming Saturday. The man does not like talent shows, nor is he willing to give up part of his Saturday to see a show he does not like. He politely declines. The Boy Scout suggests that instead he could show his support for the Boy Scouts by purchasing a few chocolate candy bars for two dollars a bar. The man purchases the bar, despite the fact that he does not like chocolate, but does like dollars. This exchange occurred because the man had received a concession by the Boy Scout. Due to this concession, the man had become indebted to the Boy Scout. The Boy Scout has readily supplied him with a means to repay that indebtedness, purchasing a few candy bars. [5]

The common element in both *reciprocity* mechanisms is that one party becomes indebted to another, and that debt must be repaid [5].

2.2.2 Commitment and Consistency

The second form of persuasion is *commitment* and *consistency*. If a person makes a *commitment* to perform an act or to support an idea, then that person is obligated to fulfill that *commitment*. The application of this type of persuasion usually involves the proposal of a deal [5]. For example, if a total stranger says to another total stranger, "If I buy you lunch, will you give me a ride home from work today." If this offer is accepted and the first person buys the second person lunch, there is now an outside source influencing the decision to drive the person home or not. Again, it is possible to see that this behavior might have arisen out of a societal need to continue the exchange of culture, ideas, and resources. Additionally, if the recipient of the lunch attempts to dodge his *commitment* to provide a ride home, the person who should receive the ride can now refer to the *commitment* that has already been made and that their lunch had already been purchased. While providing a ride may still not seem attractive to the lunch recipient, the fact that their is an existing *commitment* acts as an outside influence over their actions and decisions.

2.2.3 Liking

The third form of persuasion is *liking*. *Liking* means that people are influenced by things that are similar to them or that bring them satisfaction [5]. An example of the first application is commonly seen on television shows when a character is applying for a job. In these situations, the character's interview is not going well. Then, all of a sudden, the interviewer asks, "Where are you from?" Inevitably, the interviewer and the interviewee are from the same neighborhood,

attended the same high school, and had the same high school football coach. The course of the interview has changed, but the person being interviewed has not. The only difference now is that there is an outside influence affecting the interviewer's decisions, in this case similar life experiences.

The other form of *liking* involves behavior on the influencing party that brings about a sense of satisfaction [5]. For example, a car salesman walks up to a customer looking at a modestly priced sedan. The salesman says, "This car is not right for you. A classy gentleman such as yourself should be looking at our line of luxury sedans." A this point, the salesman has started to build up the customer's sense of satisfaction, and now the customer has an outside influence affecting his decision.

2.2.4 Authority

The fourth type of persuasion mechanism is *authority*. The main idea behind this form of persuasion is that people are influenced by the thoughts, words and actions of authority figures. *Authority* can be embodied in an individual or an organization [5]. For example, a potential customer at a car dealership is haggling with a salesman. The salesman tells the customer that he is at the lowest price he is authorized to offer. He then informs the customer that he will check with his boss, but that the answer will be the same. Two things may happen at this point. The salesman returns and tells the customer that his boss confirmed that this is the lowest price the dealership will offer, or the boss comes out and tells the customer himself. In either of these two situations, the car price remains the same and the current deal proposal has not changed. The only thing that has changed is the source of information. Regardless of how the customer proceeds from this point in the negotiation, he is influenced by this outside source.

2.2.5 Social Proof

The fifth type of persuasion is *social proof*. This type of persuasion relies on societal norms. If a person believes that societal norms apply to his current situation, then he should expect the same outcome as everyone else. In addition, societal norms are used in this form of persuasion to demonstrate how a person should act in the current situation, often referred to as the herd mentality or peer pressure [5]. The legal profession is fond of this type of persuasion. Often, a client will want to know the expected outcome of their case. Since there are many variables to consider, there is no possible way that a lawyer can guarantee an outcome. However, he may have several cases that are similar. The lawyer can then share the outcomes and circumstances with his client. Once again, nothing has changed in the situation. The people involved, the

laws, and the facts of the case all remain the same, but now an outside force is influencing the client's decisions. If the client believes that his case is similar, he is now more optimistic or more pessimistic about the outcome of his own case.

Similarly, people are influenced by societal norms in deciding how to behave or what to value [5]. A classic example from the advertising world is the Gatorade ad campaign featuring Michael Jordan and the slogan "Be like Mike." The idea behind this campaign is that Gatorade makes Michael Jordan perform at a high level on the basketball court. So, if an average or below average athlete wants to perform at a high level, they should adopt the same behavior as Michael Jordan and drink Gatorade, too. Gatorade has used this principle to influence customers to by their product. The truth of the matter is that there are many other components of his life that made Michael Jordan a great basketball player, but Gatorade has given the illusion that the norm is if a person drinks their product, they will perform better athletically.

2.2.6 Scarcity

The final form of persuasion is *scarcity*. The persuasion principle of *scarcity* is dependent on time. The person being influenced must believe that if they do not act in a certain amount of time they will miss an valuable opportunity [5]. This technique is widely used in infomercials. The infomercial presents a product and explains all the benefits of owning this product. The last thing the host does is tell the audience the price. The audience applauds with approval since the benefits of the product outweigh the monetary cost. Yet, the host is not finished. He has one more important act to perform. He tells that audience that if they act now, they will also receive an additional bonus. However, it is only available to the customers during the rest of the infomercial. If the customer waits until tomorrow, they will not receive the free bonus. Nothing has changed. The product is still the same, but now the customer is faced with the thought of losing the bonus. It does not matter what the bonus is only that they will not receive it if they wait. The customer has an outside influence affecting their decision making.

2.3 Features

Above, we discussed models of persuasion. But, what characteristics of a conversation are used to decide if persuasion is present? In machine learning, these characteristics are called "features." Before choosing an appropriate set of machine learning methods, it is important to choose a feature set that will allow for the application of these methods. In this section, we present a variety of features useful in machine learning and tools for eliminating unneeded features.

2.3.1 Word Unigrams and Bigrams

Two features commonly used in natural language processing are unigrams and bigrams. Unigrams and bigrams can be constructed from either words or characters. Character unigrams and bigrams are more appropriate for chat or blog natural language processing experiments where the post or blog belongs to the person who typed it. The use of unigrams and bigrams implies a bag of words model where the ordering is not considered in the model [6]. A unigram is comprised of single word and a bigram is a pairing of adjacent words.

2.3.2 Gappy Word Bigrams

In addition to traditional adjacent word bigrams, other types of word bigrams can be defined. One type of word bigram is the gappy word bigram. Gappy word bigrams are formed by joining word pairs that are within a given distance from each other [3]. For example, the set of gappy word bigrams produced by the phrase "the purple dog" is shown in Figure 2.1. The maximum

{START_the,START_purple,START_dog,the_purple,
the_dog,the_END,purple_dog,purple_END,dog_END}.

Figure 2.1: Gappy bigrams formed from the phrase "the purple dog"

interword distance for this set is 2. Adjacent words are considered to have a distance of 0. Therefore, gappy word bigrams with a maximum distance of 0 are equivalent to traditional word bigrams. In addition to the words in the phrase, the set of gappy bigrams includes a start of phrase marker and an end of phrase marker. The use of the these two markers allows the model to account for word occurrences at the beginning and end of a phrase.

Gappy bigrams are a variant of string kernels presented by Lodhi et al. [7]. String kernels apply to characters, not words. String kernels were shown to be effective in classifying text from a subset of the Reuters news agency dataset. Cancedda et al. applied similar principles as discussed by Lodhi et al. using words instead of characters. Cancedda et al. [8] achieved comparable results over the same data set with the added benefit of increased computational efficiency. Bikel and Sorensen [3] explored the utility of gappy n-grams with respect to sentiment detection. Bikel and Sorensen used gappy bigrams successfully to distinguish between 1-star and 5-star book reviews.

2.3.3 Orthogonal Sparse Word Bigrams

The orthogonal sparse word bigram (OSB), like the gappy word bigram, forms word pairs within a given distance. The difference lies is the treatment of the distance of the two words [9]. Using the concept of a gappy word bigram, the phrases "the purple dog" and "the big purple dog" both produce "the_dog" as member of their bigram sets. However, the distance in each phrase is different. The OSB captures this by including the distance as part of the bigram. The resulting bigram set for the phase "the purple dog" is shown in Figure 2.2.

{START_0_the,START_1_purple,START_1_dog,the_0_purple,
the_1_dog,the_2_END,purple_0_dog,purple_1_END,dog_0_END }

Figure 2.2: Orthogonal sparse bigrams formed from the phrase "the purple dog"

Using the concept of OSB, "the purple dog" and "the big purple dog" both produce OSBs that contain "the" and "dog," but they are distinct features, i.e., the_1_dog and the_2_dog.

The motivation for OSBs comes from the work of Yerazunis where *sparse binary polynomial hashing* (SBPH) was used to discriminate spam. Yerazunis achieved an accuracy of greater than 99.915% [10]. Siefkes et al. attempted the same task using OSBs, which are a proper subset of SBPH [11]. They reported similar results while dramatically reducing memory requirements. Using OSBs, Cormack et al. reported substantial improvements in bag-of-words spam filtering of short messages, including SMS messages, blog comments, and emails summary information [9].

2.3.4 Feature Discrimination

In machine learning, not all features are useful. One way to discriminate against certain features is to create and use a stop word list. Stop words are defined as words that have syntactic function, but do not contribute to the meaning of a text. One approach to generating a stop word list is to use frequently occurring words in a language [12]. Pre-compiled lists of these stop words are readily available. Figure 2.3 shows a list of English stop words that is available for download with the Natural Language Toolkit at http://www.nltk.org.

Another method of feature discrimination is to eliminate features based on the amount of information each feature contributes. It is easy to count the number of times a feature appears in a

i	me	my	myself	we	our	ours	ourselves
you	your	yours	yourself	yourselves	he	him	his
himself	she	her	hers	herself	it	its	itself
they	them	their	theirs	themselves	what	which	who
whom	this	that	these	those	am	is	are
was	were	be	been	being	have	has	had
having	do	does	did	doing	a	an	the
and	but	if	or	because	as	until	while
of	at	by	for	with	about	against	between
into	through	during	before	after	above	below	to
from	up	down	in	out	on	off	over
under	again	further	then	once	here	there	when
where	why	how	all	any	both	each	few
more	most	other	some	such	no	nor	not
only	own	same	so	than	too	very	s

Figure 2.3: Stop word list from the Natural Language Toolkit

class. With these counts, the next step is to calculate entropy using Equation 2.1. Joachims [6] and McCallum and Nigam [13] both used an entropy-base scheme to prune a feature set.

$$H(P(C|f_i)) = -\sum_j p(c_j|f_i) \log_2 p(c_j|f_i) \tag{2.1}$$

2.3.5 TextTiling

Segmentation for natural language processing can take place at a number of levels. The sentence level is useful when sentences contain the necessary features to identify a particular class. Sometimes a single sentence is insufficient. It may be necessary to look at groups of sentences or an entire paragraph. Even paragraphs may be insufficient. TextTiling was developed for exactly this reason. TextTiling groups paragraphs into tiles about a single subtopic [14]. These larger passage or tiles are then used to extract features and to perform classification experiments.

The TextTiling algorithm consists of three stages: *tokenization, lexical score determination,* and *boundary identification*. For tokenization, stop words are removed and paragraphs are marked. Text is then tokenized into pseudo-sentences to enable comparison between equal-sized strings. After tokenization, each pseudo-sentence is assigned a lexical score based on block comparison and vocabulary introduction. Finally, a depth score is assigned to each combination of possible

boundary divisions based on the two lexical scores. The final boundary selection is determined as a function of the mean and standard deviation of the depth scores.

Hearst [14] and Nomoto and Nitta [15] both report success in dividing text into single topic tiles in both English and Japanese . This suggest that TextTiling applies to a range of languages.

2.4 Machine Learning Techniques

Selecting a feature set and eliminating unhelpful features provides us with a set of inputs. The next logical step is to identify machine learning techniques that will consume these inputs. The three methods that were selected for this research are *naive bayes, maximum entropy,* and *support vector machines*. This section provides an overview of each technique, including a description of the objective function and optimization problem. These concepts will form the basis for our experimental design and analysis.

2.4.1 Naive Bayes

One machine learning method is *naive bayes*. This method uses Bayes' Rule to predict the likelihood of a class label given the features.

$$P(C|F) = \frac{P(F|C)P(C)}{P(F)} \tag{2.2}$$

It is important to note that this method has an assumption that appearance of a feature in a post is independent of the appearance of other features in the post [13]. Since this assumption is often erroneous, this method is referred to as naive. For example, *naive bayes* might be used to predict the likelihood of a post being persuasive and the likelihood of a post not being persuasive, given the features in the post. The higher of the two likelihoods is the label assigned to the post.

Another characteristic of *naive bayes* is the use of a prior probability. There are two common approaches to assigning a value to this data set. One approach is to use a uniform distribution. This means if there are four classes, then the prior probability of each class is $\frac{1}{4}$. The second approach is to use the probability of the class in the training set. If the probability of the first class is $\frac{1}{4}$ and the probability of the second class is $\frac{3}{4}$, then these will be the prior probabilities for prediction.

In order to perform a classification task using naive bayes, it is necessary to find the most

probable class, c^*. This is accomplished using the following objective function:

$$c^* = \arg\max_{c_i \in C} \left[\frac{P(F|c_i)P(c_i)}{P(F)} \right] \tag{2.3}$$

Since $P(F)$ remains constant for all classes, this function reduces to:

$$c^* = \arg\max_{c_i \in C} \left[P(F|c_i)P(c_i) \right] \tag{2.4}$$

Smoothing

Since *naive bayes* deals with probability, and it unreasonable to assume the training set will contain all features, there is a need to smooth the data in order to account for new feature values. Two commonly applied approaches are add-n smoothing and Witten-Bell smoothing [16]. Add-n smoothing requires an estimate of the size of the feature set, V. Using V, all counts for feature occurrences in the vocabulary are increased by n. The total size of the feature set for the training set is now the size of the old set, N, plus $n * V$. So,

$$P(F|C) = \frac{n}{N + n * V} \tag{2.5}$$

for all new feature values in the test set.

Witten-Bell smoothing uses a different approach. This approach uses the probability of a feature occurring to estimate the probability of new feature values. In order to estimate these probability, the number of uniques features, T, is used to estimate the probability of seeing a new type. Using V, the number of zero counts is estimated as $Z = V - T$. So, all new feature values in the test set, receive a probability according to the following equation:

$$P(F|C) = \begin{cases} \frac{count}{T+N}, \text{when } count > 0 \\ \frac{T}{Z(T+N)}, \text{ when } count = 0 \end{cases} \tag{2.6}$$

2.4.2 Maximum Entropy

Maximum entropy leverages the idea that models can be initialized to a uniform distribution and can be updated using known evidence. For any given model, the starting point is a uniform distribution over the possible number of classes. If there are 10 classes, then the probability of

each class is 0.10. However, if evidence exists that will raise the likelihood of a given class, then the model should be updated [17].

An example that illustrates the intuition behind this method is building a Spanish to English translation system modeled on a human translator. Given an English word, the initial model starts as a uniform distribution over all the words in Spanish as at first the model does not take any evidence into account. Then, the model is updated based on the number of choices in Spanish for the particular word. If there are four choices, then the probability of each Spanish word is .25. Using training data collected from a human translator, the model can now be updated to reflect that translator's use of the four words. If a particular translator uses two of the four words 65% of the time, then the new model is described the following equation:

$$P(W) = \begin{cases} P(w_1) + P(w_2) = .65 \\ P(w_1) + P(w_2) + P(w_3) + P(w_4) = 1.0 \end{cases} \tag{2.7}$$

The mathematical grounding for maximum entropy is discussed in [18, 17, 19]. Maximum entropy uses training data, D, to set constraints on a conditional distribution. These constraints take the form of real-valued functions of a document and the class, $f_i\,(d, c)$. These features are used to form an approximation of the document distribution expressed by the following equation:

$$\frac{1}{|D|} \sum_{d \in D} \sum_{c} P\,(c|d)\; f_i\,(d, c) \tag{2.8}$$

As shown in [19], when the constraints are expressed in the form above, there exists a unique distribution that has maximum entropy. The unique distribution is in the exponential form characterized by the equation below:

$$P\,(c|d) = \frac{1}{Z(d)} exp\left(\sum_i \lambda_i f_i(d, c)\right) \tag{2.9}$$

where $Z(d)$ is a normalizing factor of the form:

$$Z(d) = \sum_c exp\left(\sum_i \lambda_i f_i(d, c)\right) \tag{2.10}$$

Maximum entropy models can be subject to over-fitting, especially with sparse training data. One way to address over-fitting is to introduce a gaussian prior, λ [17]. λ can be tuned with values greater than 0. Large values of λ represent a large prior which penalizes the appearance of features, lowering their contribution to the distribution. Smaller values of λ represent a small prior and allow for closer fitting to the data [20].

2.4.3 Support Vector Machines

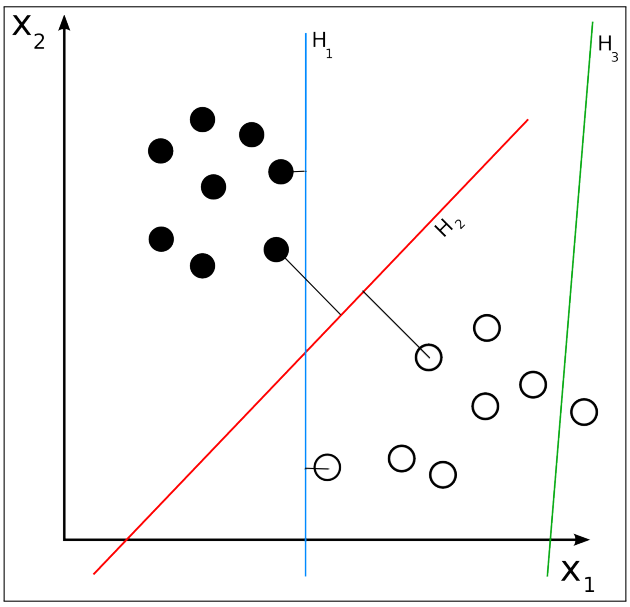

Figure 2.4: Support vector machine hyperplane determination, From [21]

A third machine learning method is the Support Vector Machine (SVM) [22, 23, 24, 25, 26, 27]. The simplest form is a binary classification. All labeled data items are represented as vectors of feature-values. If the feature vectors are of dimension n, then the task is to find a hyperplane (of $n-1$ dimensionality) to separate data points within the vector space. Figure 2.4 shows a case where more than one hyperplane exists. In order to decide between the several hyperplanes, it is necessary to find the hyperplane with the maximum margin. This hyperplane can be found by solving this optimization problem

$$\begin{aligned} \min_{w,b,\xi} \quad & \frac{1}{2} w^T w + C \sum_{i=1}^{l} \xi_i \\ \text{subject to} \quad & y_i \left(y_i w^T \phi(x_i) + b \right) \geq 1 - \xi_i, \\ & \xi \geq 0 \end{aligned} \tag{2.11}$$

where $i = 1, .., l$, $x_i \in R^n$, $y \in 1, -1^l$, and (x_i, y_i) are labeled pairs in the training set. In Figure 2.4, H_1 separates the space with a minimal margin, while H_2 separates the space with a maximum margin. In Equation 2.11, C, often referred to as cost, is used to determine the penalty for misclassification [25]. In the financial world where misclassification could result in the loss of large amounts of money, C can be set to a high value. This results in a hyperplane that minimizes loss due to classification errors. In the intelligence community, the cost may be set to a low value. This will result in a hyperplane that will tolerate higher degrees of misclassification. In this case, it is more beneficial for an analyst to see all possible documents than to try to discard some misclassified documents.

Some data sets are not linearly separable in the dimensionality of the vector space. SVM provides a mechanism for dealing with these types of problems, kernel functions. It is possible to map input data into a a dot product space, F, using a non-linear map.

$$\Phi : R^N \rightarrow F \tag{2.12}$$

This mapping may result in a high dimensional dot product that will be expensive to compute. Some kernels allow efficient computation of the dot product [28]. The following is a list of four common kernels [25]:

- linear, $K(x_i, x_j) = x_i^T x_j$
- polynomial, $K(x_i, x_j) = \left(\gamma x_i^T x_j + r\right)^d, \gamma > 0$
- radial basis function, $K(x_i, x_j) = \exp\left(-\gamma \|x_i - x_j\|^2\right), \gamma > 0$
- sigmoid, $K(x_i, x_j) = \tanh(\gamma x_i^T x_j + r)$

In the kernel functions above, γ controls the flexibility of the hyperplane. High γ values allow the hyperplane to fit the data more closely. Low γ values force the hyperplane to be more linear.

2.5 Evaluation Criteria

Regardless of which classification algorithm is selected, we must have a measure of success. Different metrics show success in different aspects of a given problem. In the following section, we present several metrics and examples of their uses.

2.5.1 Precision and Recall

Two common metrics for classification problems are *precision* and *recall*. *Precision* measures how many true positive classifications a system performed in relation to the number of total positive classifications it performed [29].

		Predicted Value	
		T	F
Actual Value	T	50	5
	F	25	20

Table 2.1: Example confusion matrix

In the example confusion matrix[30] presented in Table 2.1,

$$precision = \frac{50}{50 + 25} = .666 \tag{2.13}$$

since 50 items are classified as true positives and 25 items are classified as false positives. In a perfect world, all systems would be high precision. However, there are some times when precision is more important than others. The American legal system follows this criteria to the point that some criminals are set free, but no innocent people are imprisoned.

Recall is a metric that measures how many true positive classifications the system performed in relation to the number of actual positives in the set being classified [29]. In the example confusion matrix

$$recall = \frac{50}{50 + 5} = .909 \tag{2.14}$$

since 50 items are classified as true positives and five items are classified as false negatives. A common example from the intelligence community is that an analyst would rather see a document and discard it, rather than potentially miss a document.

2.5.2 Accuracy

Yet another metric is *accuracy*. This metric measures the number of correct classifications in proportion to the size of the set being classified [31]. In the example confusion matrix

$$accuracy = \frac{50 + 20}{100} = .7 \tag{2.15}$$

since the system classified 50 items as true positives and 20 items as true negatives. This is a useful metric in multiclass problems when the idea of a false negative or a false positive may not be strictly meaningful.

2.5.3 F-Score

One final metric is the F-score. This is the harmonic mean of the recall and precision [29].

$$\begin{aligned} f-score &= \frac{2}{\frac{1}{recall}+\frac{1}{precision}} \\ &= \frac{2}{\frac{1}{.909}+\frac{1}{.666}} \\ &= 0.769 \end{aligned} \tag{2.16}$$

The harmonic mean, in contrast to the arithmetic mean, only rewards increases in both recall and precision. If recall is increased by sacrificing precision, the F-score will fall. Similarly, if precision is increased by sacrificing recall, the F-score will fall.

2.6 Tools

2.6.1 NPSML Tools

The Naval Postgraduate School has developed a suite of tools to facilitate machine learning in its natural language processing lab. This suite of tools is publicly available via the Internet [32]. This suite of tools provides a pipelined approach to converting raw data into the NPSML format and making it useable with a variety of third-party machine learning tools. The suite also includes a naive bayes package that uses the NPSML file format.

2.6.2 Maximum Entropy (GA) Model Optimization Package

The NPSML file format is easily convertible to the Maximum Entropy (GA) Model (MegaM) optimization package file format. This makes MegaM a natural candidate to conduct maximum entropy experiments. In addition to maximum entropy, MegaM can be used to run experiments using other machine learning techniques, such as perceptron and multitron. MegaM is publicly available via the Internet [20].

2.6.3 LIBSVM

The NPSML tool suite provides a utility to convert files to LIBSVM and SVM^{light} file formats. These two third-party packages have the same file format; however, their licensing terms are

different. SVM^{light} has a stricter license, which prohibits commercial use without consent. Since this is a military institution, we decided to use LIBSVM in order to avoid any intellectual property disputes. LIBSVM is publicly available via the Internet [24].

2.7 Recent Work in Persuasion Detection

Until recently, there has been no previous work with machine learning techniques for persuasion detection. Tasks similar to persuasion detection have been explored, such as sentiment detection and perspective detection. Lin, Wiebe, and Hauptman investigated the idea of perspective identification at the sentence and document level [2]. Using the articles from the bitterlemons Web site, they were able to discriminate between Palestinian authors and Israeli authors who had written about the same topic. They used two different naive bayes methods that outperformed a support vector machine approach [2]. Bikel and Soren used machine learning techniques to differentiate between differing opinions [3]. They report an accuracy of 89% when distinguishing between 1-star and 5-star consumer reviews, using only lexical features.

Most recently, Gilbert [4] presented the first work in persuasion detection. He presented the persuasion model described in section 2.2. In [4], an annotation scheme for a persuasion corpus was presented. A pilot application of this scheme showed some agreement between annotators, but not strong agreement. After revising the annotation scheme, a more extensive study showed significant agreement between annotators. The resulting corpus of 37 transcripts were used as the basis for this thesis.

2.8 Conclusion

In this chapter, we presented all concepts relevant to detecting persuasion in conversation using machine learning techniques. We described the persuasion model used in this research. We identified features that will serve as inputs to our three machine learning classification techniques. We explained the uses of a suite metrics needed to evaluate our hypotheses. Lastly, we concluded with a presentation of software tools that enabled this research. We now have all the concepts and tools to design experiments to detect persuasion in conversation.

THIS PAGE INTENTIONALLY LEFT BLANK

CHAPTER 3:
Techniques

3.1 Introduction

In this chapter, we describe all phases of our experimental design. First, we present a description of the data. Next, we describe the process associated with making the raw data usable for machine learning, and the possible effects of the process on the results of the experiments. We, then, continue with a discussion about the features selected for the experiments. Lastly, we present the details of the experimental setup for each machine learning technique.

3.2 Description of Data

The data for these experiments comes from a corpus of labeled data created at the Naval Postgraduate School. The corpus contains negotiation transcripts annotated and adjudicated by two annotators. The tagging scheme includes all of the principles from the Cialdini model (see section 2.2). An additional category was included to capture persuasive portions of the transcripts that the Cialdini model did not capture. This category is simply labeled "Other." The corpus contains four sets of transcripts: two sets of two FBI negotiators, one set of negotiator transcripts from the Waco, Texas stand off, and a single San Diego Police negotiation. The quality of the transcripts varies, not only between the four sets, but within each set. These transcripts were transcribed from audio tapes after the events occurred. The transcribers vary in their use of punctuation, capitalization, and degree to which they attempt to capture non-English and environmental information. The raw transcripts contain 18,857 utterances, which are referred to as posts. In order to observe privacy laws and sound research principles, all data has been anonymized, with the exception of the Waco stand off transcripts, which are publicly available on the Internet.

Figure 3.1 shows the counts of each feature set by rank order. The axes in Figure 3.1 are $\log - \log$ axes. Since the shapes of the graphs are roughly linear, each feature set follows a power-law distribution, commonly called a Zipf law [33].

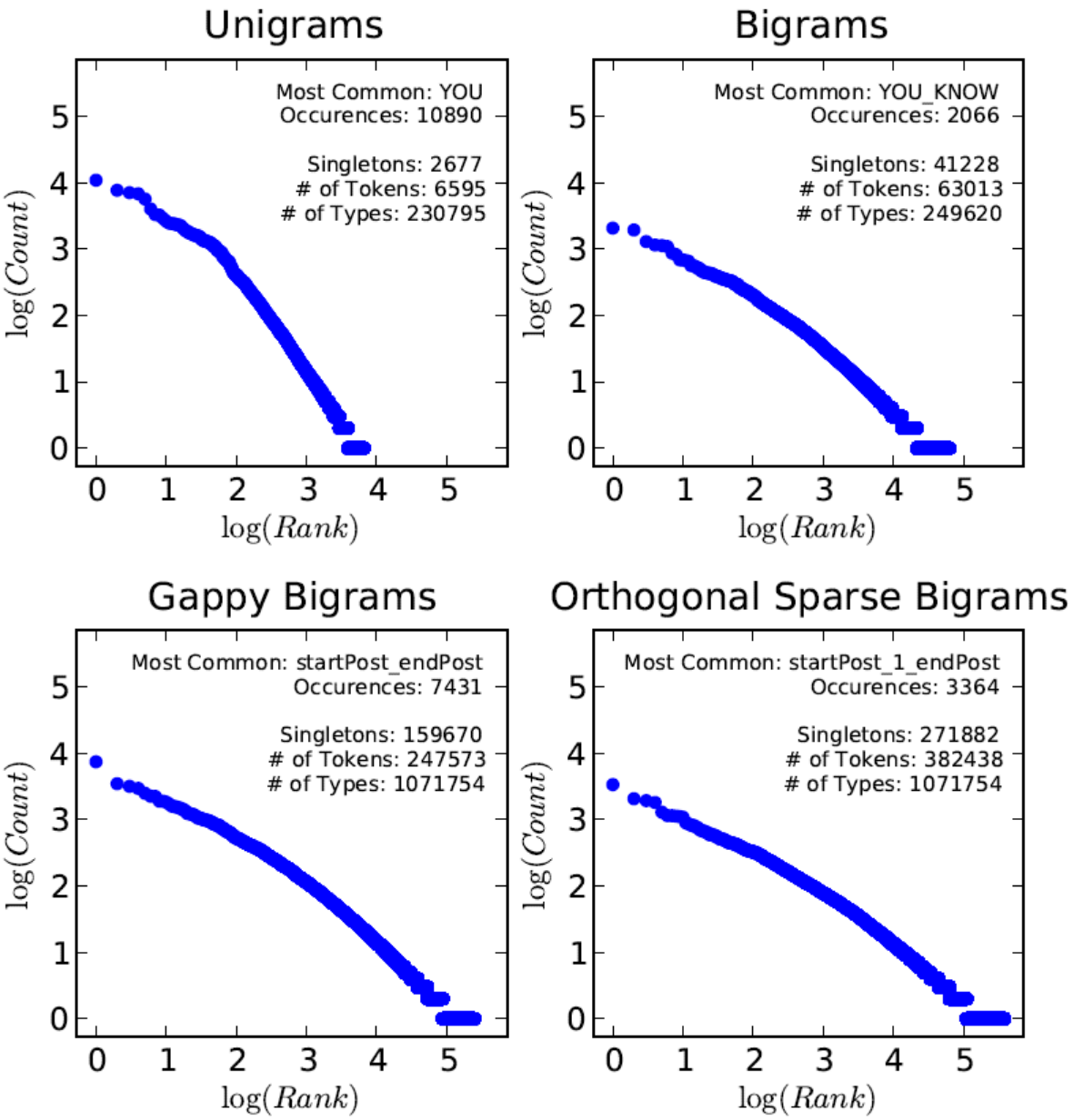

Figure 3.1: Scatter plots of four feature sets

Tables 3.1 through 3.4 show the features with the most predictive power by class for each feature set using naive bayes. Note that some of the most predictive features occur in both classes. In order to address this phenomenon, we apply a feature discrimination technique described in section 3.14.1. Tables 3.3 and 3.4 show that posts shorter than five words are more likely to be non-persuasive.

Unigrams			
persuasive		not persuasive	
$Feature$	$-\ln(\text{p}(Feature))$	$Feature$	$-\ln(\text{p}(Feature))$
KNOW	4.858649	OKAY	3.609763
DONT	5.519006	KNOW	3.667643
GET	5.551796	<NAME>	3.892805
OKAY	5.585698	YEAH	4.077149
GONNA	5.620789	WELL	4.08324
IM	5.734118	IM	4.189884
GOT	5.908471	DONT	4.250603
<NAME>	6.008555	RIGHT	4.25543
MINUTES	6.062622	GET	4.433245
RIGHT	6.180405	THATS	4.585952

Table 3.1: Predictive unigrams, ranked by class (NLTK stopwords removed)

Bigrams			
persuasive		not persuasive	
$Feature$	$-\ln(\text{p}(Feature))$	$Feature$	$-\ln(\text{p}(Feature))$
YOU_KNOW	5.190261	startPost_OKAY	4.951206
GOING_TO	6.061165	YOU_KNOW	5.208624
startPost_WELL	6.180557	startPost_YEAH	5.282209
startPost_OKAY	6.393786	OKAY_endPost	5.489037
TO_DO	6.564516	startPost_I	5.499077
TALK_TO	6.569849	startPost_WELL	5.622425
YOU_TO	6.569849	YEAH_endPost	5.914684
AND_I	6.575211	I_DONT	5.936323
IF_YOU	6.580602	startPost_NO	5.976169
I_DONT	6.586022	GOING_TO	6.077023

Table 3.2: Predictive bigrams, ranked by class

Gappy Bigrams			
persuasive		not persuasive	
$Feature$	$-\ln(\mathrm{p}(Feature))$	$Feature$	$-\ln(\mathrm{p}(Feature))$
YOU_KNOW	6.430376	startPost_endPost	4.90189
YOU_YOU	6.488989	startPost_YOU	5.898575
YOU_TO	6.520223	startPost_I	5.911928
TO_YOU	6.567135	YOU_endPost	6.045993
startPost_YOU	6.632002	startPost_OKAY	6.197869
I_YOU	6.772881	TO_endPost	6.298162
YOU_endPost	6.843388	YOU_KNOW	6.451672
AND_YOU	6.861407	THE_endPost	6.455523
YOU_AND	6.984368	startPost_YEAH	6.554118
startPost_I	7.054029	OKAY_endPost	6.566976

Table 3.3: Predictive gappy bigrams, ranked by class

Orthogonal Sparse Bigrams			
persuasive		not persuasive	
$Feature$	$-\ln(\mathrm{p}(Feature))$	$Feature$	$-\ln(\mathrm{p}(Feature))$
YOU_0_KNOW	6.865789	startPost_1_endPost	5.814345
GOING_0_TO	7.736692	startPost_2_endPost	6.4359
startPost_0_WELL	7.856084	startPost_0_OKAY	6.485967
I_1_YOU	7.889236	YOU_0_KNOW	6.743384
YOU_3_YOU	8.042763	startPost_0_YEAH	6.81697
startPost_0_OKAY	8.069313	startPost_3_endPost	6.912361
YOU_4_YOU	8.115192	startPost_4_endPost	6.947836
TO_1_YOU	8.13415	OKAY_0_endPost	7.023797
YOU_1_TO	8.188218	startPost_0_I	7.033837
YOU_1_YOU	8.224211	startPost_0_WELL	7.157186

Table 3.4: Predictive OSBs, ranked by class

3.3 Raw Data to Usable Text

The original data format was 4 excel workbooks with tabs. Each tab in the workbook was a single transcript. The original excel type was .xlsx. The first step was to save these workbooks as .xls to be readable by python scripts. After changing the file format to .xls, a python script easily read each cell. By iterating over the rows, it was possible to make a simple one-for-one copy and save the ASCII text to a comma-separated value file.

In addition to converting the format, the scripts also cleaned up irregularities in the data. While reading from each cell, the scripts made several changes to each post. The punctuation was removed as these features were generated by the transcriber, not the participants in the negotiation. As this research is concerned with persuasion attempts by the negotiations participants and not identification of transcribers, those features were discarded. During the anonymization process, named entities received bracketed place holders, <HOSTAGE TAKERS FIRST NAME> for example. These named entity place holders were replaced with single tokens by replacing all spaces between angle brackets with underscores. The resulting token from the previous example is <HOSTAGE_TAKERS_FIRST_NAME>. Lastly, transcribers introduced more features by placing comments in the text in an effort to capture non-English and environmental information. Since individual transcribers varied in their use of comments and since this research is focused on text, the role of comments was minimized by using a similar approach to the named entities. All comments appear as single tokens with square brackets and underscores.

The last step before writing the clean post to the .csv file was to add some additional information before each post. The first field in each line is the transcript name. The second field is the line number from the .xls file. These two pieces of information proved invaluable during the data preparation tool development phases. Without this data, it would have been very difficult to verify errors is data conversion. It is also hoped that this information will be of use during future analysis. Lastly, the class label and speaker were added before the post. Each transcript was saved into its own separate .csv file. Figure 3.2 shows the resulting file format.

3.4 Additional Segmentation

Each entry in the .csv files represents a post. Some posts are only a few words, while others are several sentences long. This variance in length may affect the classification task. Therefore, it is necessary to explore another type of segmentation. One way to achieve a different type of segmentation is to group related posts together. For this, we used the TextTiling method

Rogan_beta,221,1,ON80,Yeah but that fiddler isnt gonna cost so much if you walk out easy

Rogan_beta,223,1,ON80,come on <HT01> youre just making it worst on yourself

Rogan_charlie,641,0,PNl,Alright [both_hang_up]

Rogan_charlie,691,3,HT1,Bring <Wife_First_Name> and Ill come out

Figure 3.2: Example from .csv files

described in section 2.3.5. Each post is treated as a paragraph. Using the Natural Language Toolkit (NLTK), each .csv was grouped into a series of segments, referred to as tiles. The format of the .tile file is similar to the .csv file format. The first field is the transcript name. The second field is the line range from the original .xls file. Line ranges are inclusive and appear in the following format: <first line number>_<last line number>. The line number range is followed by the a 0 or 1 to indicate not persuasive or persuasive. A tile was labeled as persuasive if any of the original posts were labeled with any one of the nine persuasion categories. A tile was labeled as not persuasive if all of the original posts were labeled as not persuasive. The last field in each line in the file is the tile itself. Figure 3.3 shows the resulting file format.

Rogan_beta,210_231,1,[Sighs] Go have a cup of coffee I [...] Yeah but that fiddler isnt gonna cost so much if you walk out easy [...] come on <HT01> youre just making it worst on yourself [...] Well you didnt get caught [Laughs] No but I did this time Yeah you did this time

Rogan_charlie,638_654,1,Alright OK Alright Alright [both_hang_up] [...] nothing they gonna do Aint nobody gonna shoot you Throw the gun off the balcony

Rogan_charlie,677_687,0,<ON1> I be I be in a police car right behind [...] But I tried to tell <Wife_First_Name> she should have told me about that guy right

Figure 3.3: Example from .tile files

3.5 Initial Cross Validation

The next step in the data preparation pipeline was to divide the data into test and training sets. First, all .csv files were concatenated into a single .csv file and all .tile files were concatenated into a single .tile file. These files maintained the original post and tile ordering within each transcript. These files were each internally shuffled prior to creating test and training sets. We used ten-fold cross validation. This means there were 10 test and training splits for each type

of segmentation. Each test set was 10% of the number of post or tiles. The other 90% was used for training data. Each post and each tile appeared in only one of the 10 test sets. Each post and each tile appear in 9 of the 10 training sets. In creating models for each machine learning technique, training sets were used to train each model. Then, each model was tested by classifying the data in the test set using the model. No item in the test set was used for training the model. For each type of segmentation, the data was split into test and training files that were paired using indexes, meaning that test.0 and train.0 were paired for a single experiment. These files were stored in the original .csv or .tile format.

3.6 Randomized Grouping Cross Validation

The method of randomization presented in section 3.5 had the potential to influence the outcome of the parameter tuning experiments due to repetition of similar phrases within a single transcript. A review of the data showed that often similar phrases appeared shortly after or well after the first appearance of the phrase. This is quite common in this domain for two reasons. The first reason is that many of the conversations were conducted using poor communications equipment or were conducted in noisy enviroments where the speakers were forced to repeat themselves. The second reason for this behavior is the nature of a negotiation; each party is trying to achieve certain goals. Failure to achieve that goal does not result in abandoning the goal. On the contrary, if the goal is important, such as releasing a hostage, the request to have that goal met is more likely to be repeated later in the conversation.

As a result of this repetition, it was necessary to develop a new approach to randomization. The file names for the each transcript were shuffled and grouped into 6 groups of 6 transcripts. Each group of 6 transcripts was concatenated into a single test set. The training sets were formed by concatenating 5 test sets and pairing it with the 6th test set. This process was conducted for both posts and tiles. The shortest transcript (19 posts, 0 tiles) was not included in these folds.

3.7 Leave-One-Out and Leave-One-In Validation

The two methods presented in section 3.5 and section 3.6 were used to find suitable parameters for each machine learning technique and to determine whether persuasion detection is possible. If persuasion detection is possible, it is necessary to investigate whether or not these methods generalize. Since no other data sets are available, we used two approaches. The first way, called

leave-one-out, entailed training on three transcript types and testing on the remaining type. The second method, called leave-one-in, entailed training on only one transcript type and testing on the other three types.

3.8 Differences in Transcript Type

In the course of this research, the results indicated that one group of transcripts might be sufficiently different from the others as to adversely affect the results of the experiment. In order to observe the effects of this group, leave-one-out validation and leave-one-in validation were repeated using the three remaining transcript types. The same procedures were used as are outlined in section 3.7.

3.9 Majority and Single Classifier Voting

The results of this research indicated that a voting scheme over the three types of classifiers could be beneficial. The first voting scheme is called majority voting. In order to be classified as persuasive, a post had to receive two or more votes from the three classifier. A second scheme called single classifier voting was also explored. In this scheme, a post was classified as persuasive if it received a single vote from any of the three classifiers. For both schemes, all votes were counted only within a feature set. Voting schemes that combined votes across the different feature sets were not explored.

3.10 Feature Extraction

As noted earlier in this section, the original data contained many features introduced by the transcribers and not by the negotiation participants. After removing these artifacts, the only features that remained are words. Since unigrams and bigrams are common features used in other natural language processing research, we used them as a baseline to compare the performance of other features. In this research, we used two additional feature types: gappy word bigrams and orthogonal sparse word bigrams, as described in section 2.3. These features were chosen based on their performance in sentiment detection and short message spam filtering. In addition, these two features are less expensive with regards to memory and to computational complexity than similar, more verbose features, such as sparse binary polynomial hashing [10].

For each test and training file, each of the four features are extracted from each tile or post. The result is a file in the NPSML format as shown in Figure 3.4. The key field was the transcript name and the line number or range. The weight was set 1.0 for all posts and all tiles. The classes

key␣weight␣class␣feature_label_1␣feature_value_1[␣feature_label_2␣feature_value_2...]\n

Figure 3.4: Feature extraction file format

were restricted to 0 for non-persuasive and 1 for persuasive. During extraction, all letters are converted to uppercase. This collapses "lowercase" and "Lowercase" into a single feature, for example. This also helped reduce the number of false features introduced by transcribers. Gappy bigrams and orthogonal sparse bigrams were extracted using a distance of 4. Since adjacent words have a distance of zero, this means that a single word was paired with each of the 5 closest words. No stemming was done. Figure 3.5 shows the resulting file entries for a single post. Each feature type was extracted and saved into a file using the following naming

Taylor5_LP_2124 1.0 0 WELL 1 I 1 CAN 1 SEE 1 HOW 1 YOU 1 DID 1

Taylor5_LP_2124 1.0 0 startPost_WELL 1 WELL_I 1 I_CAN 1 CAN_SEE 1 SEE_HOW 1 HOW_YOU 1 YOU_DID 1 DID_endPost 1

Taylor5_LP_2124 1.0 0 startPost_WELL 1 startPost_I 1 startPost_CAN 1 startPost_SEE 1 startPost_HOW 1 WELL_I 1 WELL_CAN 1 WELL_SEE 1 WELL_HOW 1 WELL_YOU 1 I_CAN 1 I_SEE 1 I_HOW 1 I_YOU 1 I_DID 1 CAN_SEE 1 CAN_HOW 1 CAN_YOU 1 CAN_DID 1 CAN_endPost 1 SEE_HOW 1 SEE_YOU 1 SEE_DID 1 SEE_endPost 1 HOW_YOU 1 HOW_DID 1 HOW_endPost 1 YOU_DID 1 YOU_endPost 1 DID_endPost 1

Taylor5_LP_2124 1.0 0 startPost_0_WELL 1 startPost_1_I 1 startPost_2_CAN 1 startPost_3_SEE 1 startPost_4_HOW 1 WELL_0_I 1 WELL_1_CAN 1 WELL_2_SEE 1 WELL_3_HOW 1 WELL_4_YOU 1 I_0_CAN 1 I_1_SEE 1 I_2_HOW 1 I_3_YOU 1 I_4_DID 1 CAN_0_SEE 1 CAN_1_HOW 1 CAN_2_YOU 1 CAN_3_DID 1 CAN_4_endPost 1 SEE_0_HOW 1 SEE_1_YOU 1 SEE_2_DID 1 SEE_3_endPost 1 HOW_0_YOU 1 HOW_1_DID 1 HOW_2_endPost 1 YOU_0_DID 1 YOU_1_endPost 1 DID_0_endPost 1

Figure 3.5: Feature extraction file example for unigrams, bigrams, gappy bigrams, and orthogonal sparse bigrams

convention shown in Figure 3.6.

3.11 Additional Pre-computed Information

In addition to extracting the features from the data, it was necessary to extract token counts and token entropy. Each of which were calculated for both posts and tiles. Each training set had three count files and one entropy file associated with it. The three count files contained the feature counts for each class and the overall feature counts. From these count files, a single

<feature abbreviation>_NPSML_<test|train>.<0-9>	
File Name	File Description
BGM_NPSML_train.0	Bigrams, NPSML format, training set 0
GBG_NPSML_train.0	Gappy bigrams, NPSML format, training set 0
OSB_NPSML_test.0	Orthogonal sparse bigrams, NPSML format, test set 0
UGM_NPSML_test.0	Unigrams, NPSML format, test set 0

Figure 3.6: Feature extraction naming convention

entropy file was generated. These files were used for file format conversion, as well as, feature discrimination.

3.12 Classification Tasks

We focused on binary classification of posts from the original data and tiles developed from those posts. In the original data, each post was labeled with the type of persuasion from Cialdini's persuasion model with an additional "Other" category. However, due to the sparseness of persuasion at the post level (less than 12% of posts are persuasive), we decided to address the binary classification of "persuasion" versus "not persuasion." Possible refinement of this task are discussed in section 5.2.

3.13 Parameter Tuning

The following section is a description of the experimental process conducted for three machine learning techniques presented in Chapter 2. In each section, there is a discussion of the the tools and parameter tuning process used.

3.14 Naive Bayes

Naive bayes experiments were conducted using a naive bayes package developed in the Naval Postgraduate School Natural Language Processing Lab. The learning portion of this package used an NPSML file as input and generated a model which was written out as a binary file. The binary representation of the model eliminated round off errors that could have occured due to ASCII representations of floats. The learning portion implemented Laplace add-one smoothing. The classification portion of this package used the model generated from the learning process. The input for the classification process was an NPSML file. The resulting output was a 2 column text file listing the key and the predicted class.

The parameter tuned in this set of experiments was the prior probability of the two classes. The information for the prior was contained in the model output by the learning portion of the package. After the learning phases, a binary input-output tool developed in the Naval Postgraduate School Natural Language Processing Lab was used to overwrite the two existing values. The prior probability of persuasion was increased by 5% for each new set of experiments. The resulting set of experiments included one set with the prior probability proportional to probability of the class in the training set and 19 experiments with the prior probability of persuasion set a multiple of 5%, starting at 5% and ending at 95%. For each experiment the prior probability of not persuasion was set to the $1-p(\text{persuasion})$. For each feature type, a series of 20 experiments was conducted over ten folds.

3.14.1 Feature Discrimination

In addition to tuning the prior probability of persuasion, experiments were also conducted using a reduced feature set. The feature set for these experiments was reduced based on the entropy files described earlier in this chapter. These files were sorted based on their entropy. Features with the highest entropy appeared first. Features with the same entropy appeared in reverse alphabetical order. Experiments were run removing the top 5%, 10%, 15%, 20%, 25%, 30%, 35%, 40%, 45% and 50% of the types in the feature set from the training and test data. These experiments followed the naive bayes protocol previously outlined. For each feature type, a series of 200 experiments was conducted over ten folds and six folds.

3.15 Maximum Entropy

Maximum entropy experiments were conducted using the Maximum Entropy GA Model (MegaM) Optimization package developed at the University of Utah [20]. The learning portion of this package used a file format similar to the NPSML file format. NPSML files can be converted to MegaM format by removing the first two columns (key and weight). The learning portion of this package used a MegaM file as input and generates a model which by default was written to the standard output. The standard output can then be piped to a file. The resulting model was a 2 column text file listing features and their weights. Weights describe the λ_i's in Equation 2.9. When running the learning portion of these experiments the following command was used:

megam -quiet -fvals -lambda λ -repeat 100 binary train.i $>$ weights.i , where i is an index

Figure 3.7: MegaM command

The -quiet flag suppressed output to the screen. The -fvals flag signified the use of named features, as opposed to an integer index to a feature list. The -lambda flag allowed the gaussian prior in the model to be tuned. The -repeat flag ensured that iterative improvement is attempted at least 100 times. This is needed because the MegaM documentation warned that sometimes the algorithm stops prior to convergence. The binary flags indicated what type of model to build.

The parameter tuned in this set of experiments was the gaussian prior, referred to as λ. The initial value of λ was set to 2^{-10}. For each subsequent set of experiments over the 10 folds, λ was increased by a power of 2. The last set of experiments used $\lambda = 2^{10}$. Higher values of λ result in a smoother fitting distribution.

3.16 Support Vector Machine

SVM experiments were conducted using the LIBSVM package developed at the National Taiwan University. The learning portion of this package used a file format shown in Figure 3.8. Conversion from the NPSML format to the LIBSVM format required a dictionary that mapped

class␣feature_index_1:feature_value_1[␣feature_label_2:feature_value_2...]\n

Figure 3.8: LIBSVM file format

the features indexes to human readable feature names. The count files created during data preprocessing were used for this task. The NPS Machine Learning library contained a tool to convert NPSML files to LIBSVM files.

The learning portion of this package used an LIBSVM file as input and generated a model which was written to a file. The resulting model was a list of weights and support vectors for the hyperplane dividing the two classes. When running the learning portion of these experiments the following command was used: The -q flag suppressed output to the screen. The -cost flag

svm-train -q -cost C -gamma γ train.i model.i , where i is an index

Figure 3.9: LIBSVM command format

allowed the penalty for misclassification in the model to be tuned. The -gamma flag allowed the γ in the radial basis kernel to be tuned. Cost and γ are described in section 2.4.3.

The parameters tuned in this set of experiments were cost and γ. The initial value of cost was set to 2^{-5} and increased by 2 powers of two until reaching a maximum value of 2^{15}. For each value

of cost, the initial value of γ was set to 2^{-15} and increased by 2 powers of two until reaching a maximum value of 2^5 [25]. Each cost-γ pair was used for experiments across 10 folds and 6 folds.

3.17 Conclusion

This chapter has presented a description of the data, the process associated with data usable for machine learning, the features selected for the experiments, the details of the experimental setup for each machine learning technique. Now that the experimental design process is clear, the next step is to review and analyze the results.

THIS PAGE INTENTIONALLY LEFT BLANK

CHAPTER 4:
Results and Analysis

4.1 Introduction

In this chapter, we present the results of our experiments. First, we present and discuss the results of our parameter tuning experiments. Next, we describes the results of the experiments conducted using 6-fold cross-validation over post and tiles. We continue with an exploration of the effects of holding out transcript types and training on singular transcript types. Lastly, we explore the possibility that one set of transcripts is significantly different from the rest.

4.2 Naive Bayes Parameter Tuning

As discussed in Chapters 2 and 3, naive bayes, maximum entropy, and support vector machines all have parameters than can affect the outcome of an experiment. This being so, it is important to find the correct parameters for the experiments conducted in this research. This was done by conducting a grid search over the parameter space. The results of these parameter tuning experiments are presented in section 4.5.

For the naive bayes experiments, the two parameters in question are the prior probabilities of the classes and the percentage reduction of the number of tokens in the feature set. If a class has a high prior probability, the prior probability may overwhelm the conditional probabilities in the argmax function presented in Equation 2.4. This may lead to overprediction of the more frequently occurring class. Conversely, more equally distributed classes will have similar prior probabilities. This may lead to over emphasis of the occurrence of the tokens in a post or a tile. The prior probabilities based on occurrence in the entire data set are shown in Table 4.1. The results of the parameter tuning experiments show that increasing the prior probability of persuasion increased the F-scores for experiments over posts and tiles. This change in F-score resulted from an increase in recall with a much smaller decrease in precision.

Naive Bayes		
Class	Posts	Tiles
Persuasive	0.116	0.455
Not Persuasive	0.884	0.545

Table 4.1: Prior probabilites of each class by segmentation type

The parameter tuning experiments showed the effects of feature selection on F-scores, as well. Features were removed based on their conditional entropy. The features with the highest entropy were removed first. The highest F-scores for post experiments were achieved using no reduction in the features set for unigrams and OSBs and a 10% reduction for bigrams and gappy bigrams. The maximum F-scores for tile experiments required more pruning of the feature set. Unigrams performed best over tiles with a 40% reduction. Bigrams and OSBs experiments performed best with a 50% reduction, while gappy bigrams only required a 10% reduction. The results of these experiments appear in Figures 4.1 through 4.16. Since tiles are longer than posts,it was more likely that the same features were repeated in different post. This resulted in more features with minimal amounts of information. Removing these feature resulted in the maximum F-scores for each set of experiments.

Table 4.2 shows the parameters used for all subsequent naive bayes experiments. Note that some parameter sets performed equally well during parameter tuning (see Figures 4.1 through 4.16).In these cases, the prior probability closest to the probability of occurrence in the data set and the smallest percentage of reduction in the feature set were chosen.

Naive Bayes				
	Posts		Tiles	
Features	Prior	Reduction	Prior	Reduction
Unigrams	0.15	0.00	0.95	0.40
Bigrams	0.45	0.10	0.95	0.50
Gappy	0.95	0.10	0.65	0.10
OSBs	0.95	0.00	0.95	0.50

Table 4.2: Naive bayes parameters

4.2.1 Naive Bayes over Posts

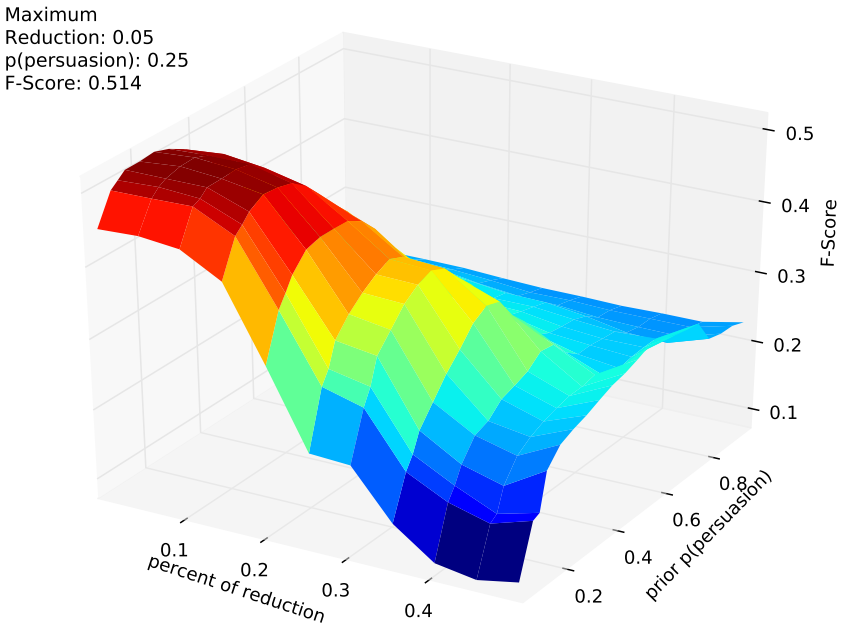

Figure 4.1: Naive bayes 10-fold averaging for unigrams over posts

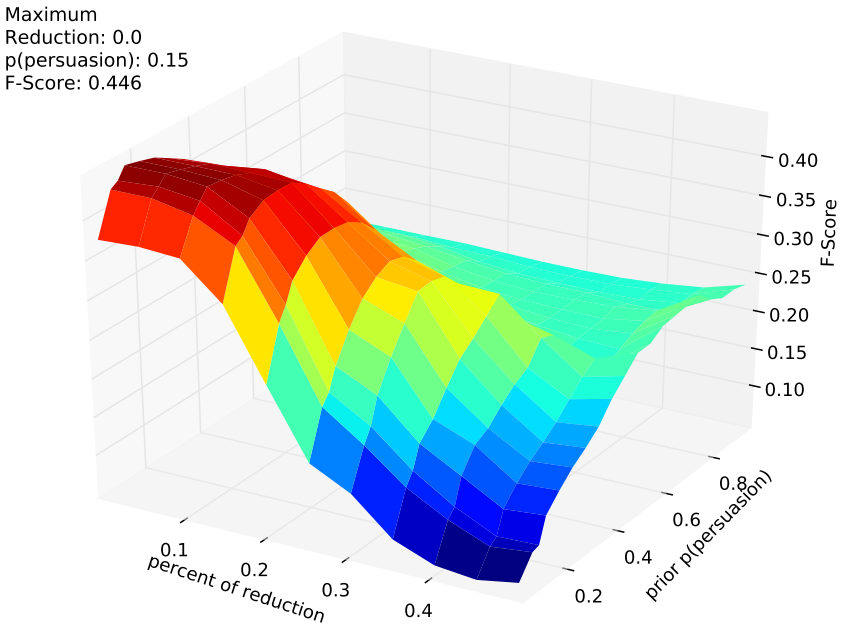

Figure 4.2: Naive bayes 6-fold averaging for unigrams over posts

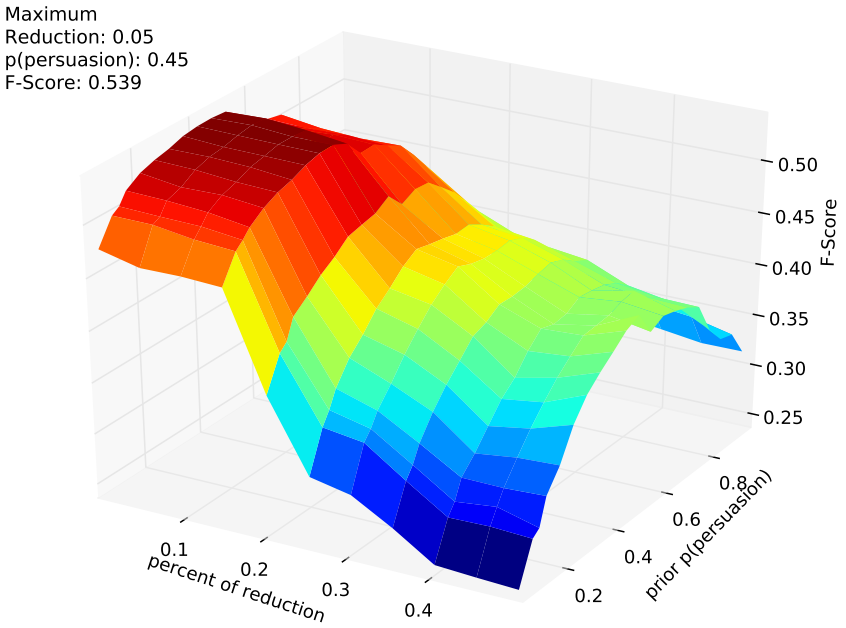

Figure 4.3: Naive bayes 10-fold averaging for bigrams over posts

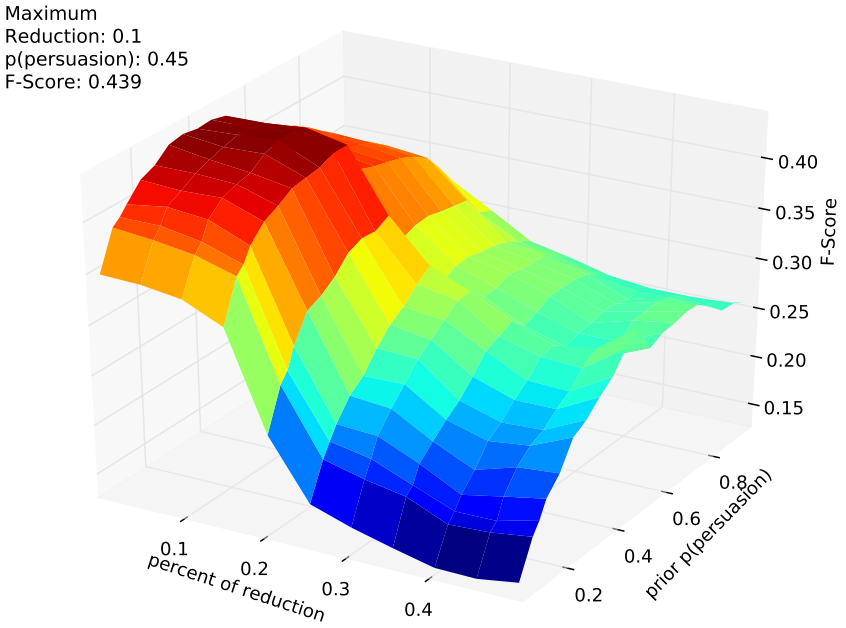

Figure 4.4: Naive bayes 6-fold averaging for bigrams over posts

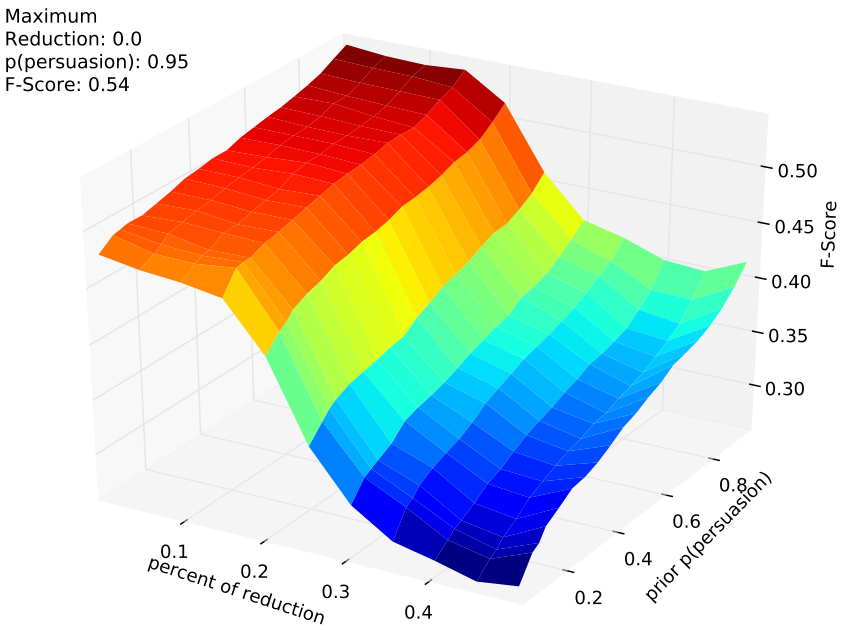

Figure 4.5: Naive bayes 10-fold averaging for gappy bigrams over posts

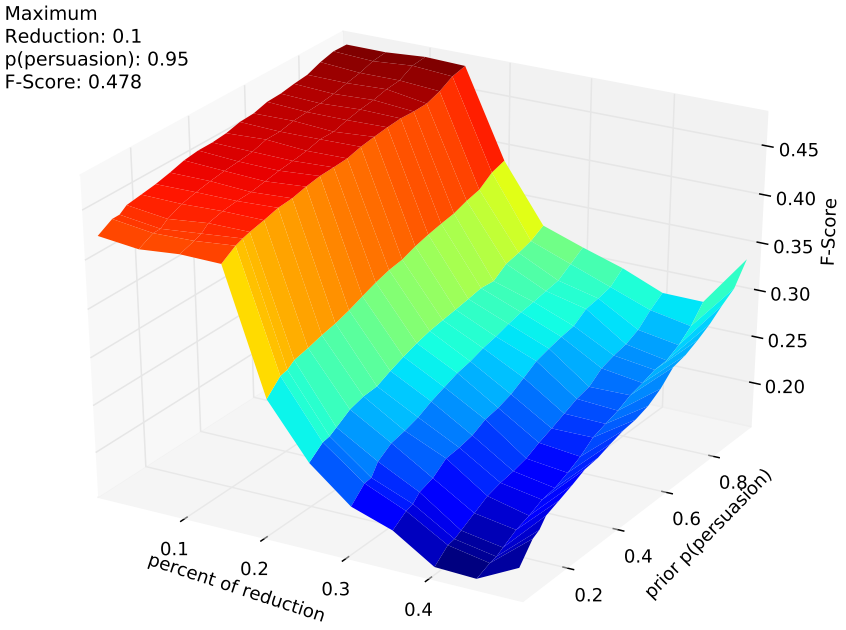

Figure 4.6: Naive bayes 6-fold averaging for gappy bigrams over posts

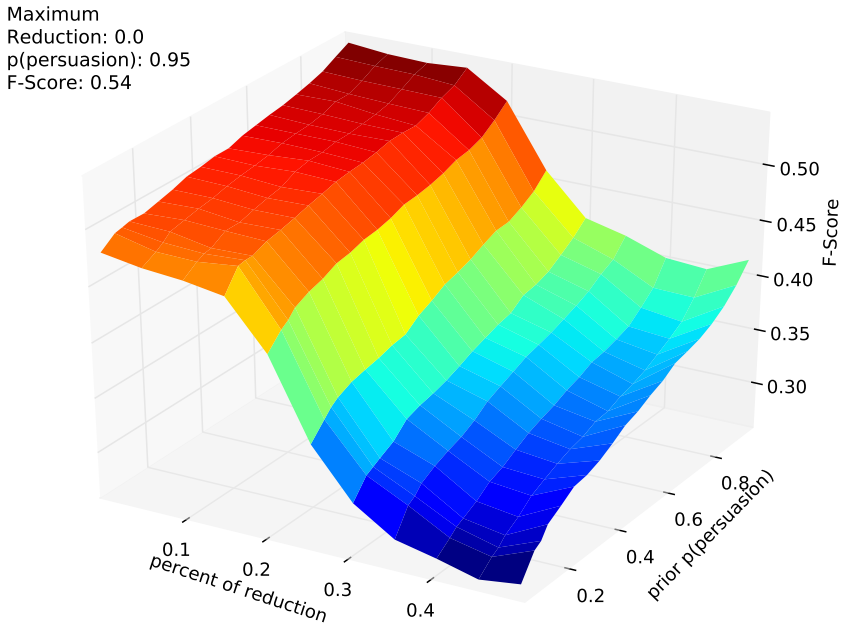

Figure 4.7: Naive bayes 10-fold averaging for orthogonal sparse bigrams over posts

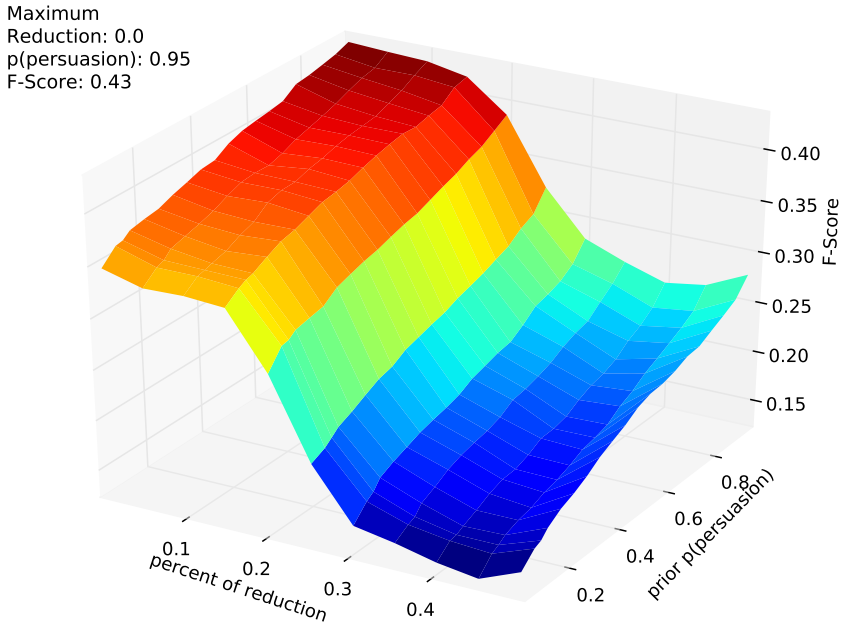

Figure 4.8: Naive bayes 6-fold averaging for orthogonal sparse bigrams over posts

4.2.2 Naive Bayes over Tiles

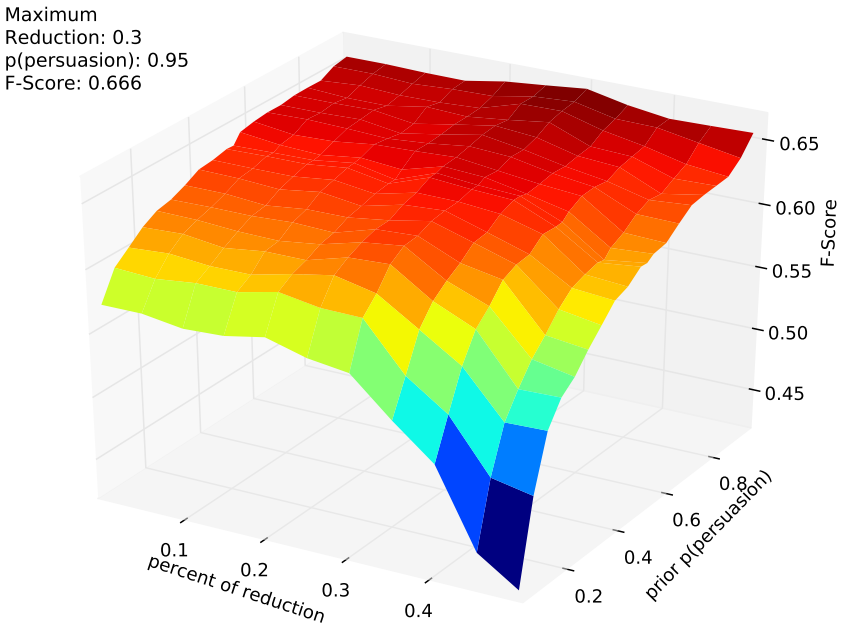

Figure 4.9: Naive bayes 10-fold averaging for unigrams over tiles

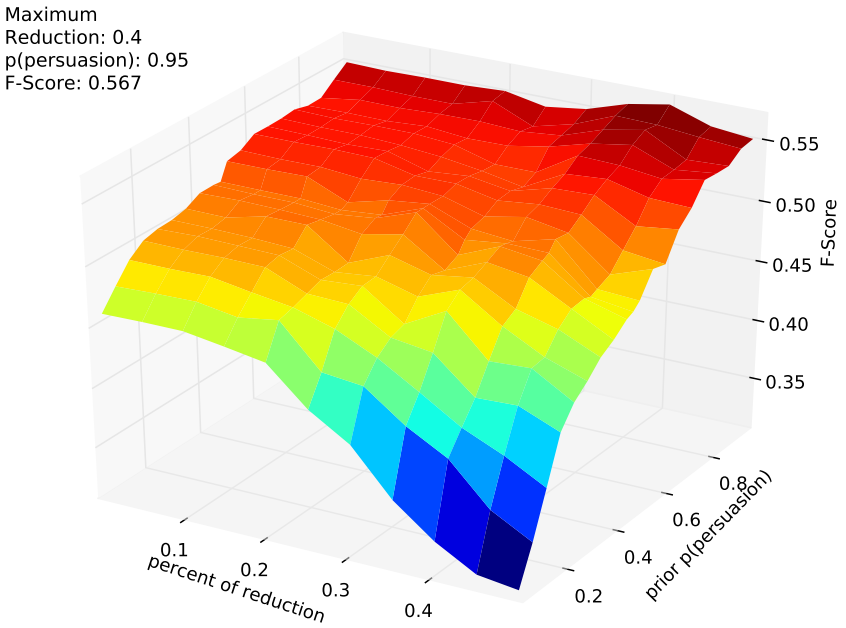

Figure 4.10: Naive bayes 6-fold averaging for unigrams over tiles

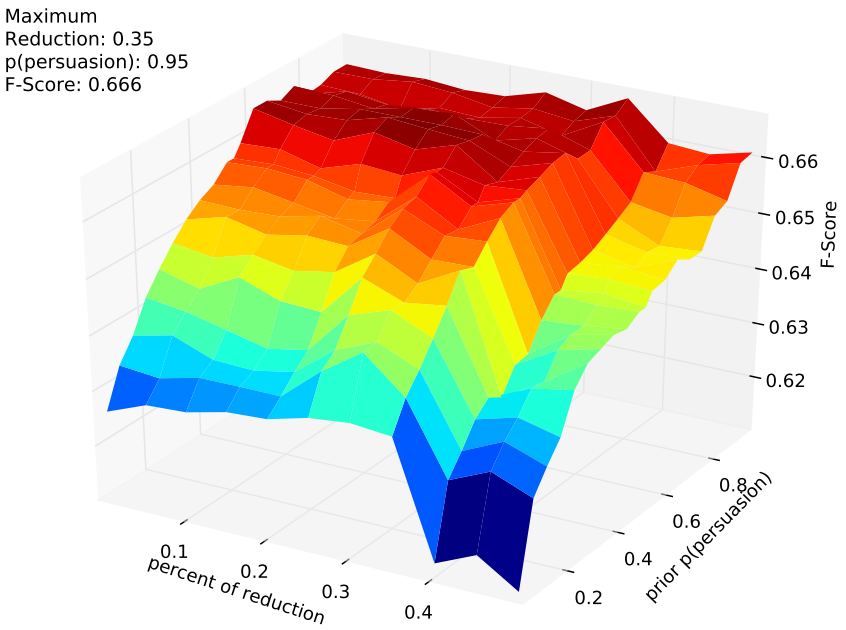

Figure 4.11: Naive bayes 10-fold averaging for bigrams over tiles

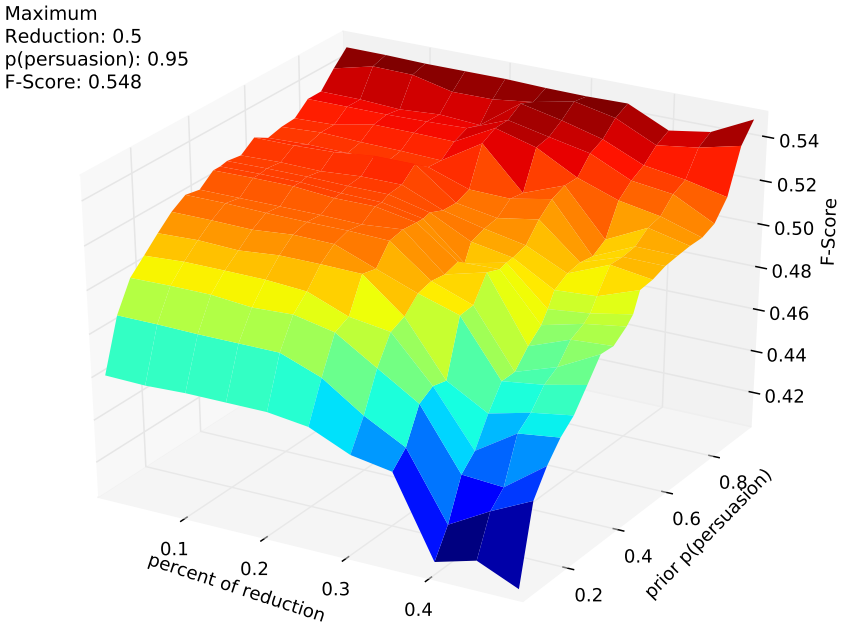

Figure 4.12: Naive bayes 6-fold averaging for bigrams over tiles

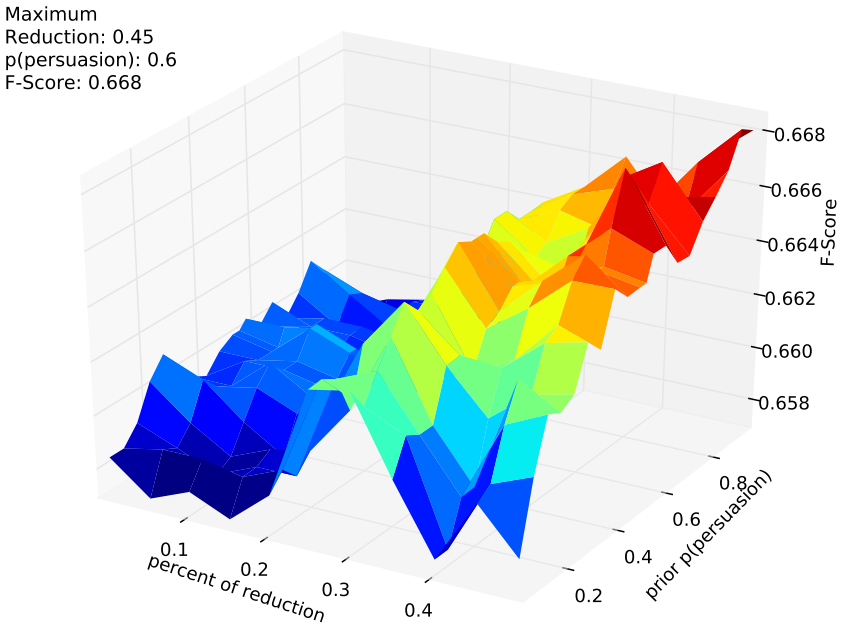

Figure 4.13: Naive bayes 10-fold averaging for gappy bigrams over tiles

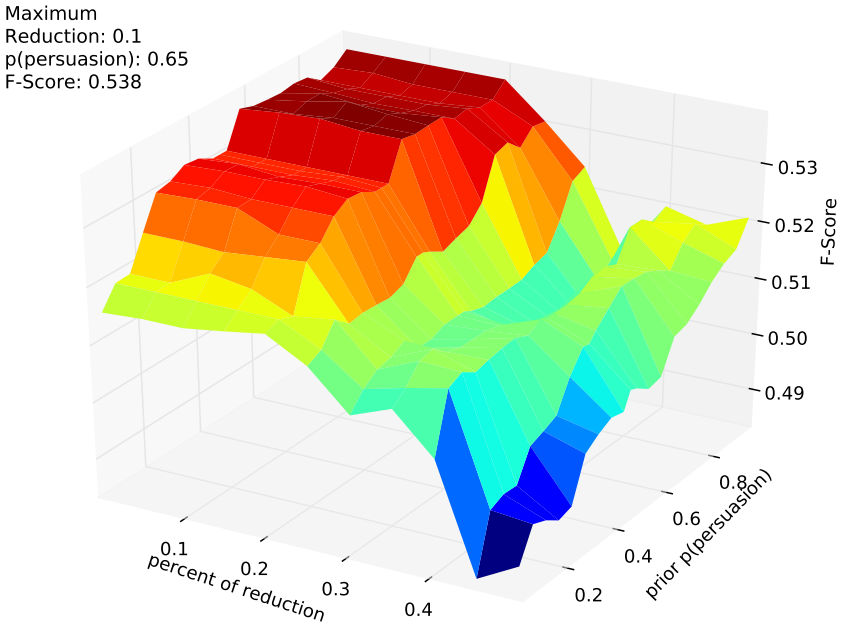

Figure 4.14: Naive bayes 6-fold averaging for gappy bigrams over tiles

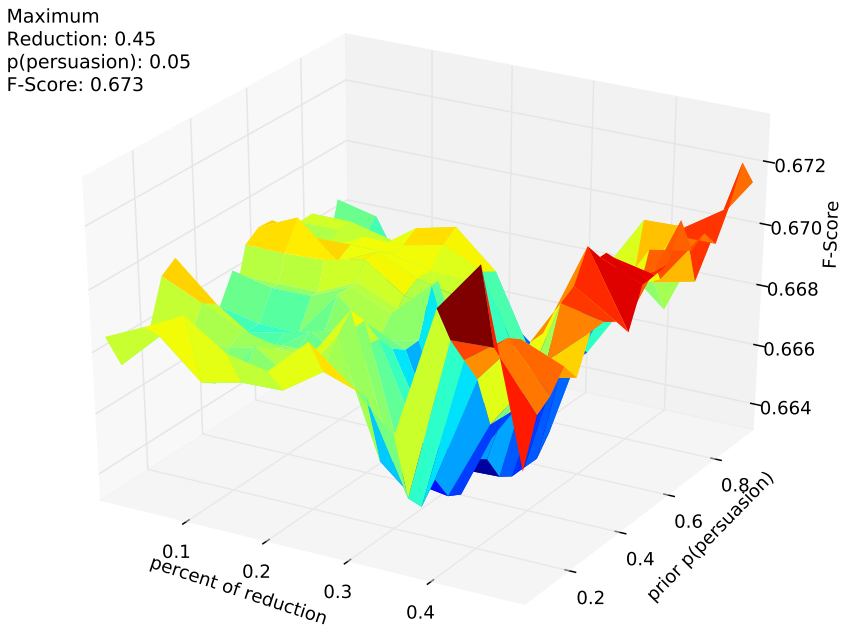

Figure 4.15: Naive bayes 10-fold averaging for orthogonal sparse bigrams over tiles

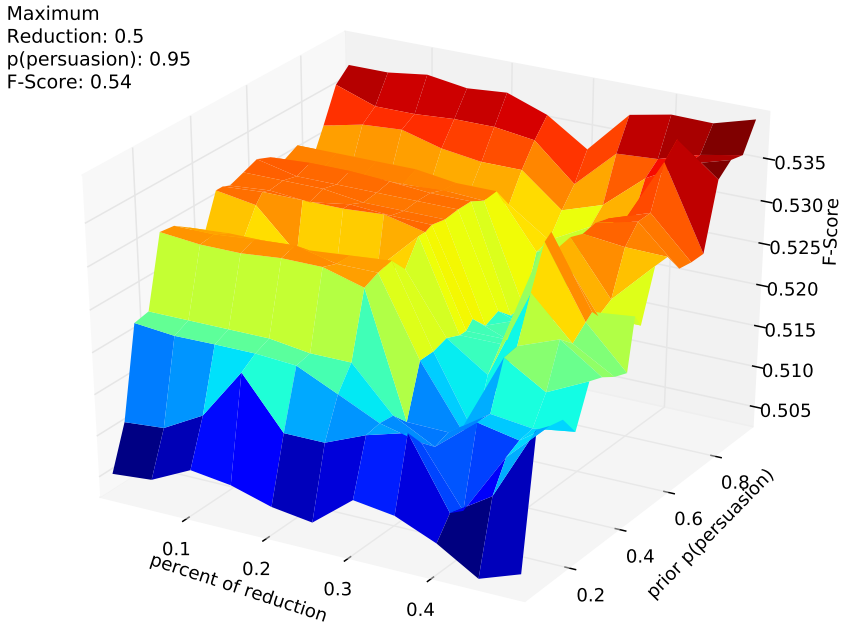

Figure 4.16: Naive bayes 6-fold averaging for orthogonal sparse bigrams over tiles

4.3 Maximum Entropy Parameter Tuning

For the maximum entropy experiments, the two parameters in question are the gaussian prior, λ, and the percentage reduction of the number of tokens in the feature set. A high λ places a higher penalty on the weights of the features. This leads to a smoother fitting of the distribution to the data. A low λ penalizes the weights of the features less, and leads to a tighter fitting of the distribution to the data. The results of the parameter tuning experiments in Figures 4.17 through 4.32 show that decreasing the gaussian prior probability increased the F-scores for experiments over posts. This was primarily due to steep decreases in recall as the value of λ was increased. The maximum F-scores for tile experiments were achieved by increasing the gaussian, with the exception of unigrams. The results of the parameter tuning experiments are shown in Figures 4.17 through 4.32. For bigram experiments, the F-score trend was explained by an increase in recall with a less significant increase in precision as the value of λ was increased. For increased λ values, the unigram experiments showed an initial decrease in recall, which was then followed by an increase in recall.

As in the naive bayes experiments, features were removed based on their conditional entropy (see Equation 2.1). The features with the highest entropies first were removed first. The highest F-score for post experiments were achieved using no reduction for unigrams and bigrams, a 5% reduction for OSBS, and 10% reduction for gappy bigrams. Tile experiments performed best with a 10% reduction for unigrams and bigrams and a 5% reduction for gappy bigrams and OSBs.

Table 4.3 shows the parameters used for all subsequent maximum entropy experiments. As in the naive bayes experiments, note that some parameter sets performed similarly. In these cases, the largest λ value was chosen to discourage over fitting [17]. The smallest percentage of reduction in the feature set were also chosen.

Maximum Entropy				
	Posts		Tiles	
Features	Lambda	Reduction	Lambda	Reduction
Unigrams	2^{-2}	0.00	2^{-7}	0.10
Bigrams	2^{-2}	0.00	2^{8}	0.10
Gappy	2^{-10}	0.10	2^{10}	0.05
OSBs	2^{-8}	0.05	2^{10}	0.05

Table 4.3: Maximum entropy parameters

4.3.1 Maximum Entropy over Posts

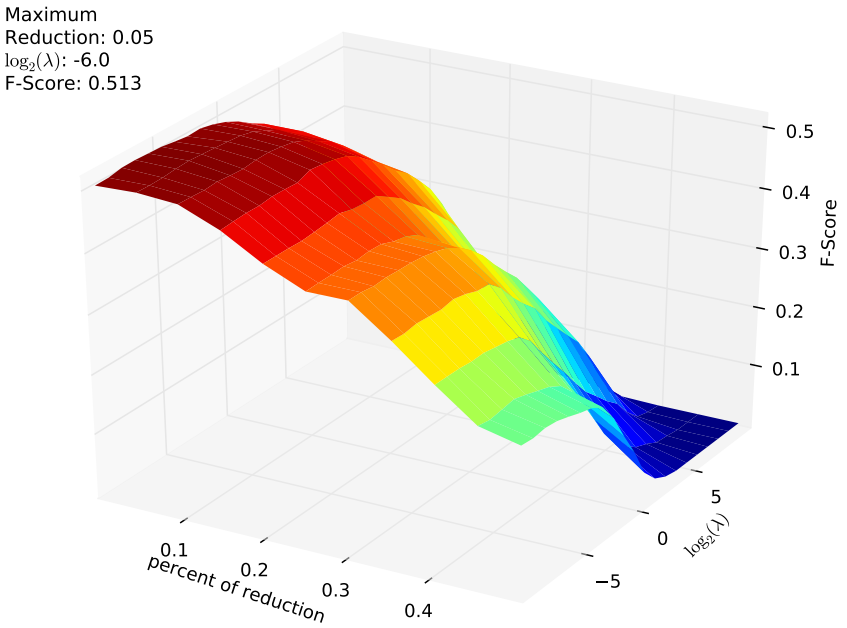

Figure 4.17: Maximum entropy 10-fold averaging for unigrams over posts

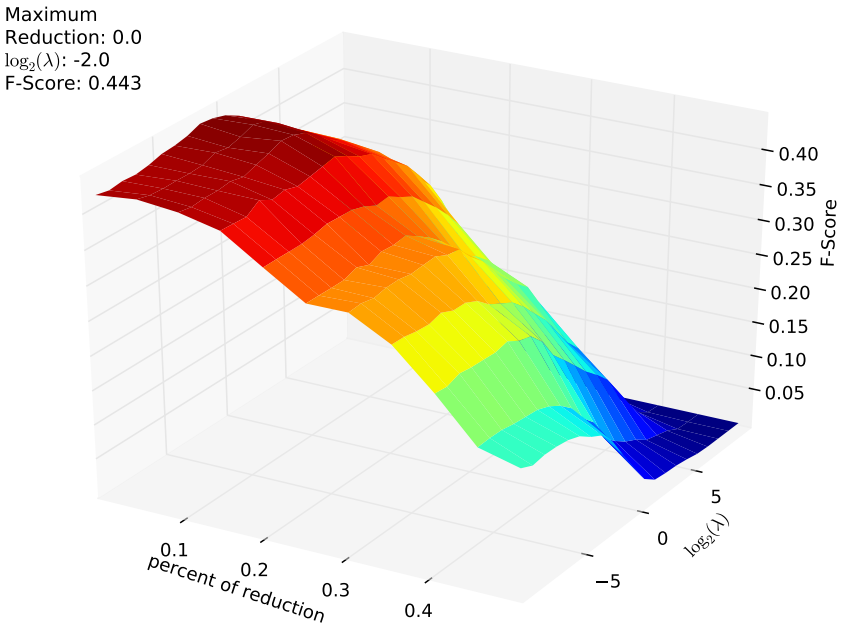

Figure 4.18: Maximum entropy 6-fold averaging for unigrams over posts

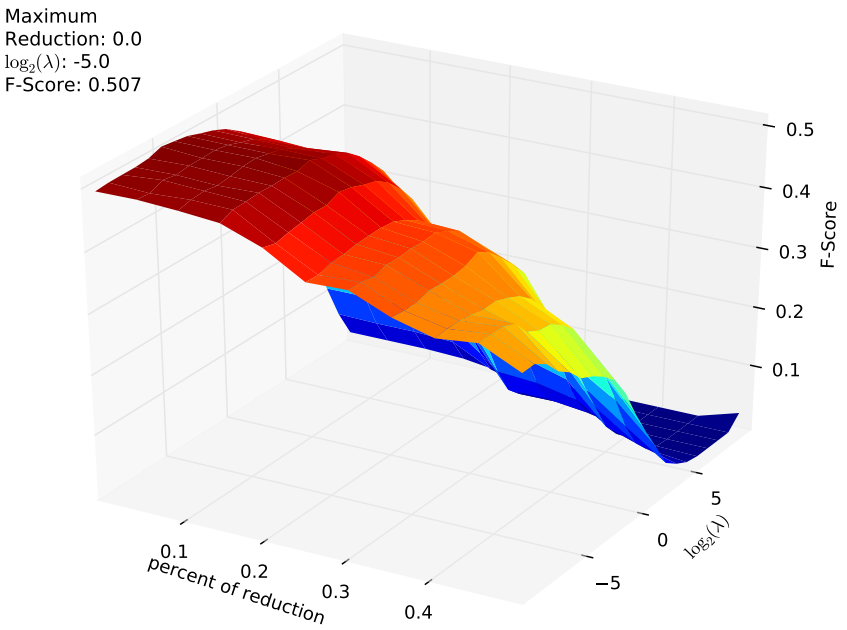

Figure 4.19: Maximum entropy 10-fold averaging for bigrams over posts

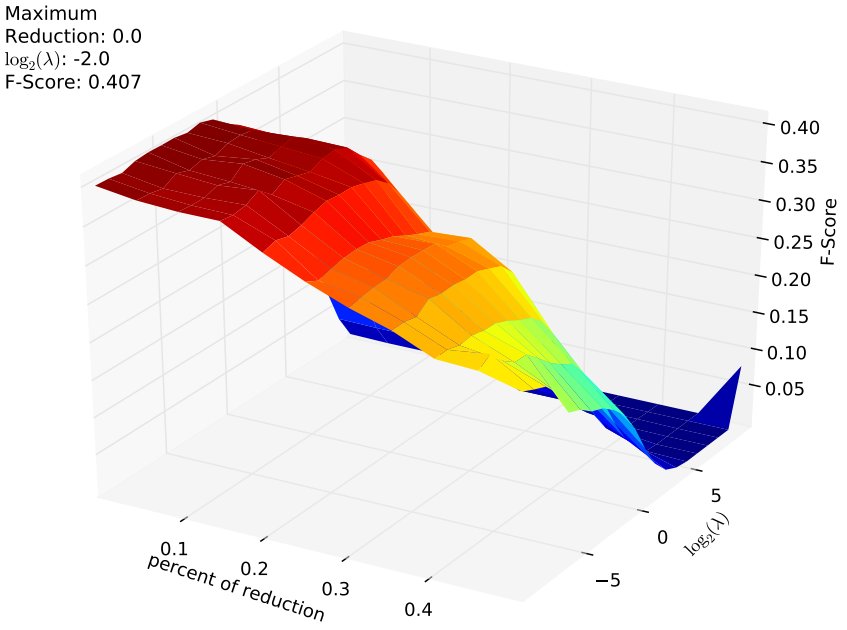

Figure 4.20: Maximum entropy 6-fold averaging for bigrams over posts

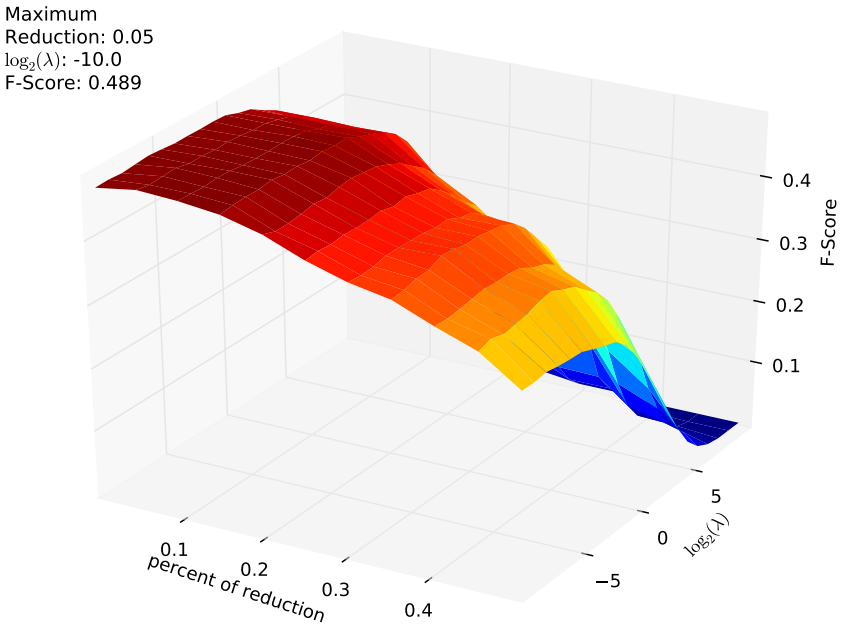

Figure 4.21: Maximum entropy 10-fold averaging for gappy bigrams over posts

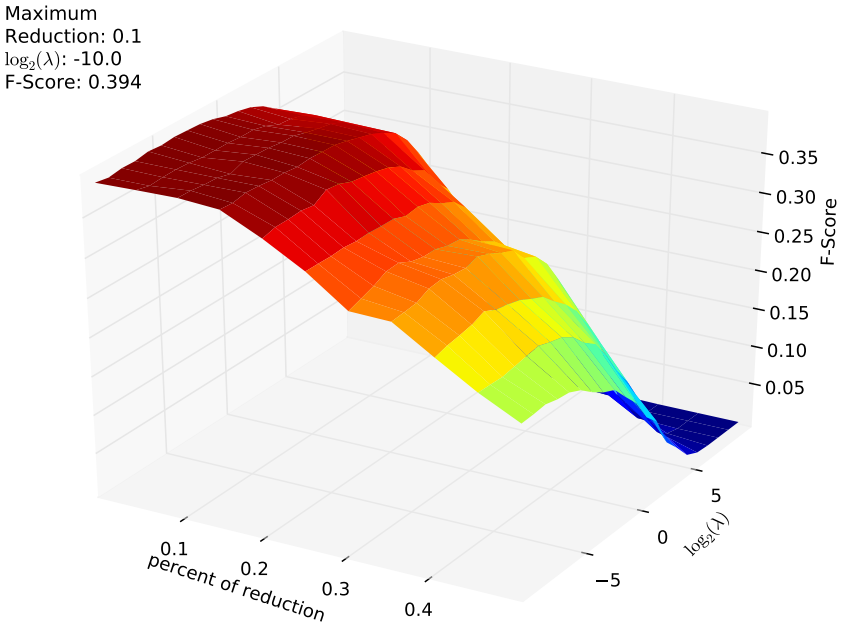

Figure 4.22: Maximum entropy 6-fold averaging for gappy bigrams over posts

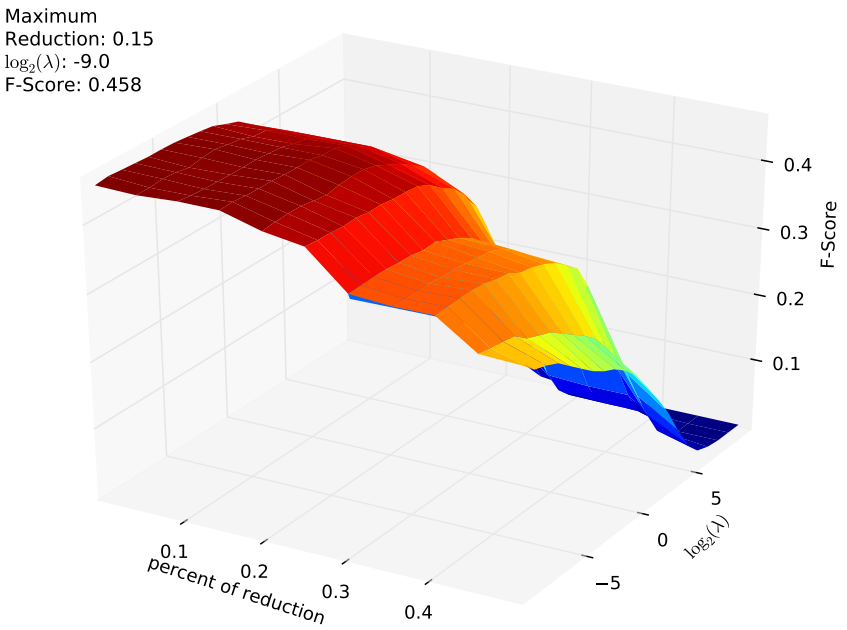

Figure 4.23: Maximum entropy 10-fold averaging for orthogonal sparse bigrams over posts

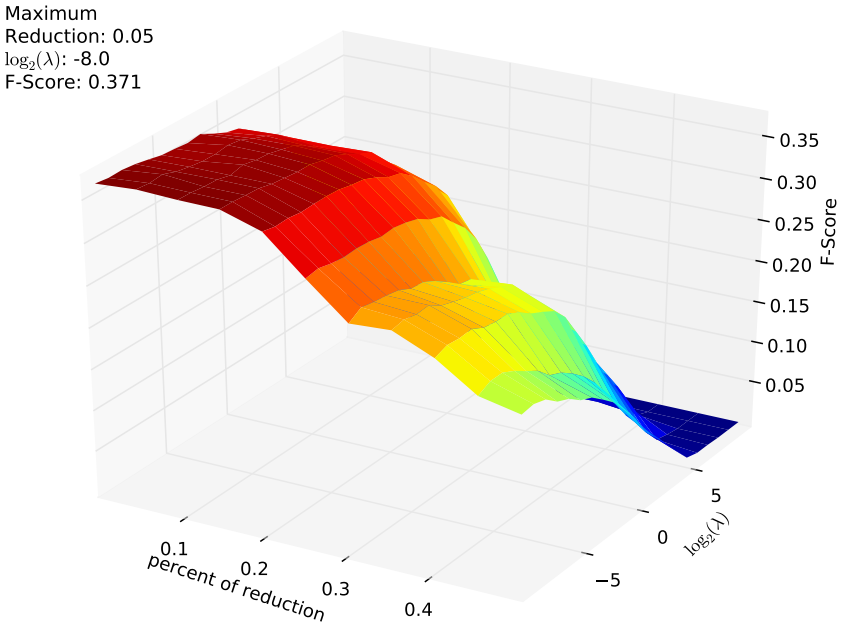

Figure 4.24: Maximum entropy 6-fold averaging for orthogonal sparse bigrams over posts

4.3.2 Maximum Entropy over Tiles

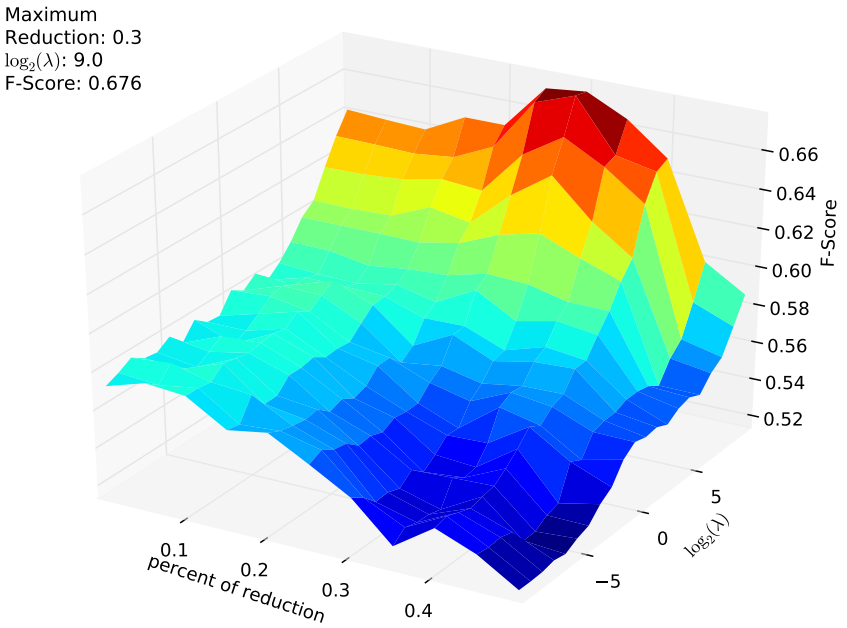

Figure 4.25: Maximum entropy 10-fold averaging for unigrams over tiles

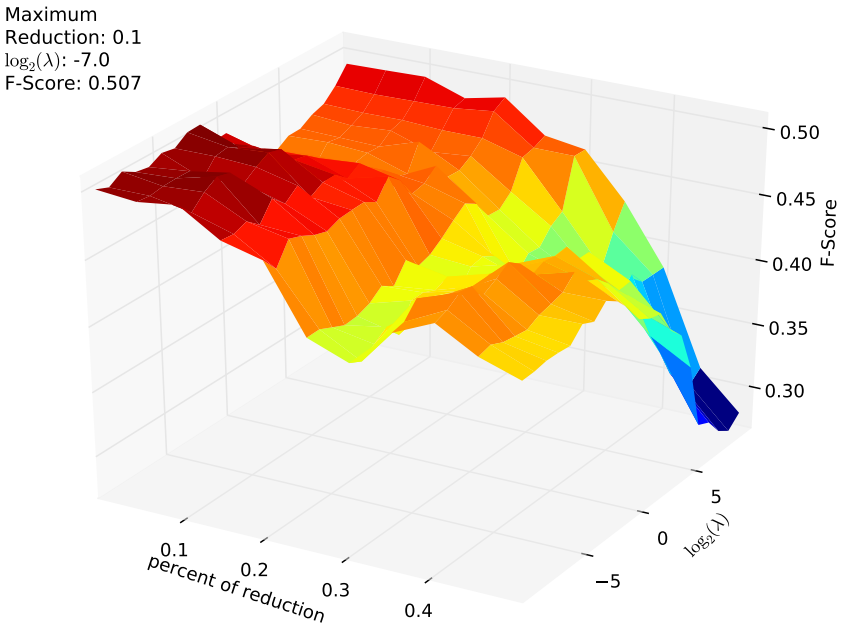

Figure 4.26: Maximum entropy 6-fold averaging for unigrams over tiles

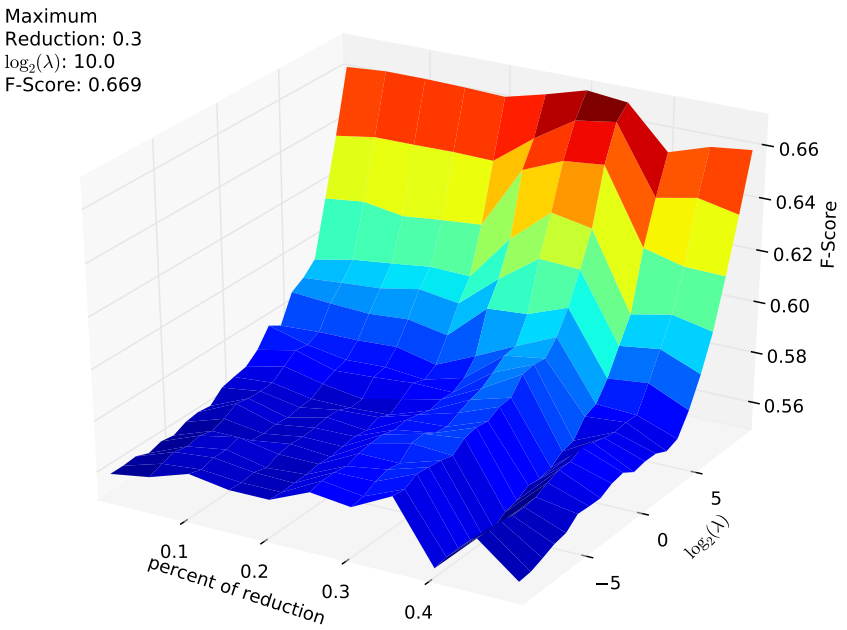

Figure 4.27: Maximum entropy 10-fold averaging for bigrams over tiles

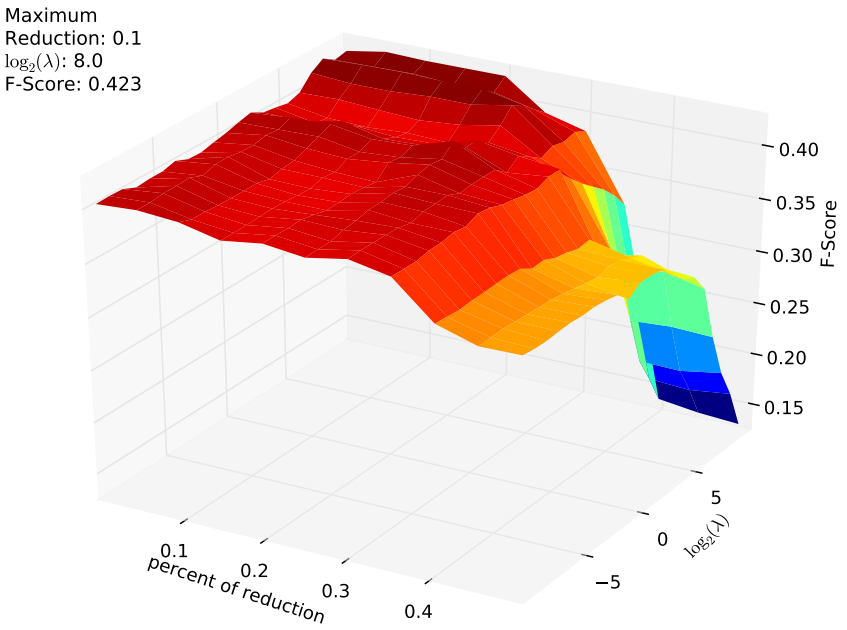

Figure 4.28: Maximum entropy 6-fold averaging for bigrams over tiles

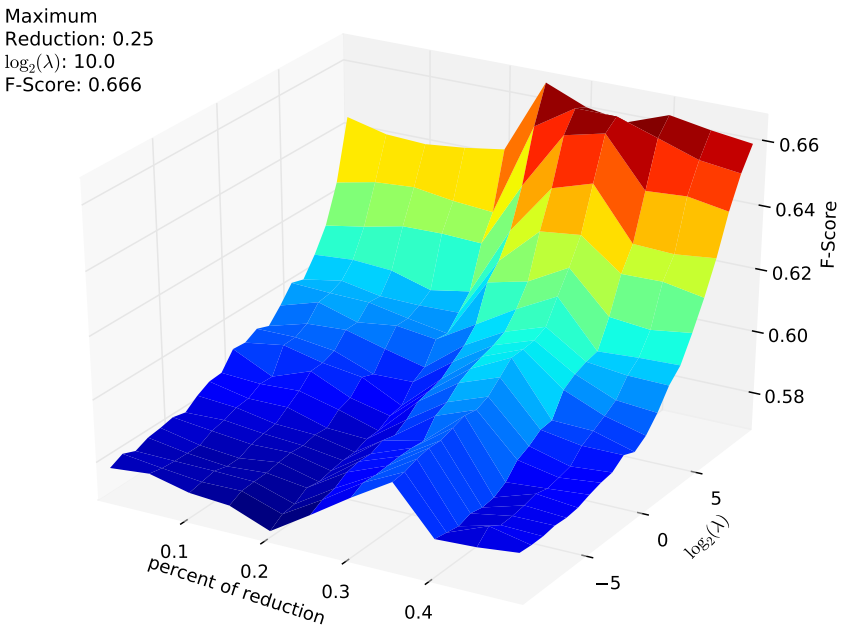

Figure 4.29: Maximum entropy 10-fold averaging for gappy bigrams over tiles

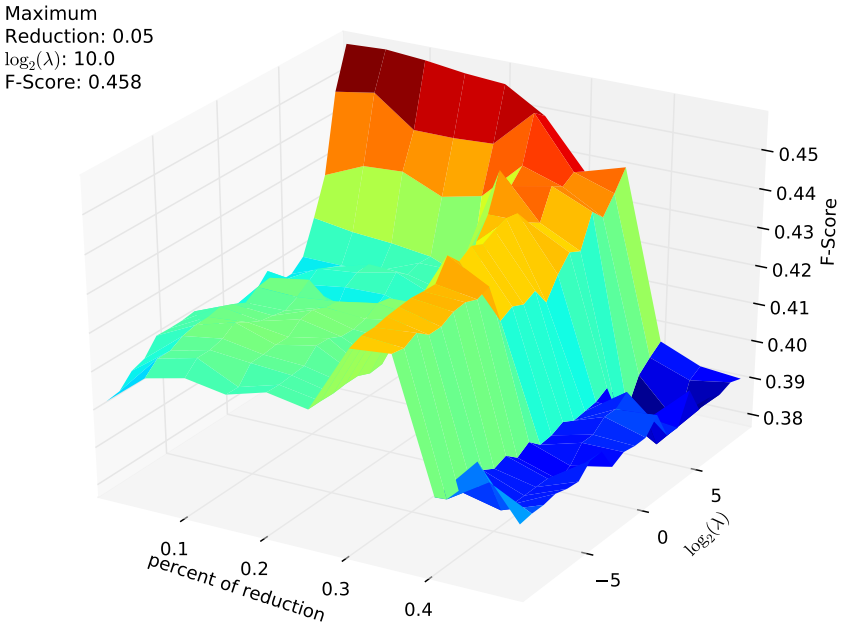

Figure 4.30: Maximum entropy 6-fold averaging for gappy bigrams over tiles

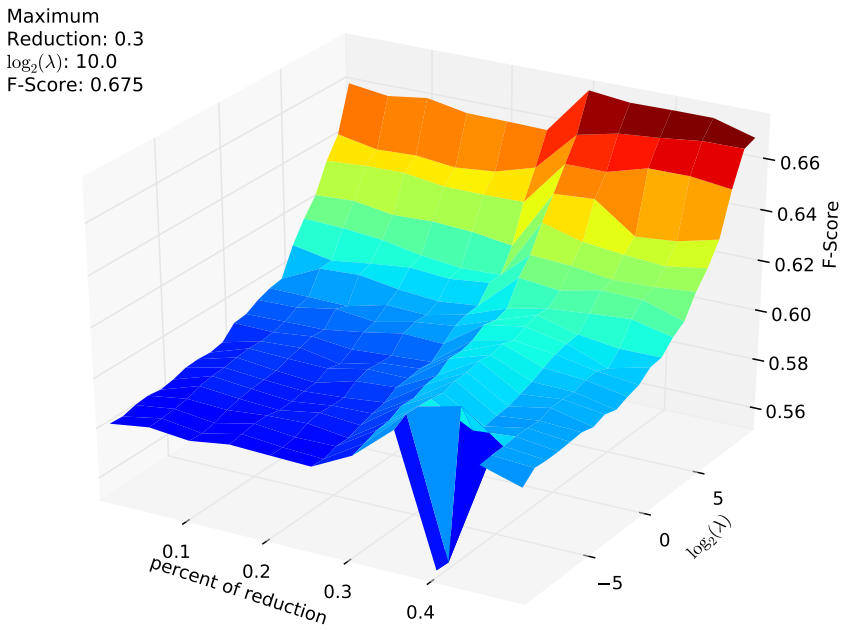

Figure 4.31: Maximum entropy 10-fold averaging for orthogonal sparse bigrams over tiles

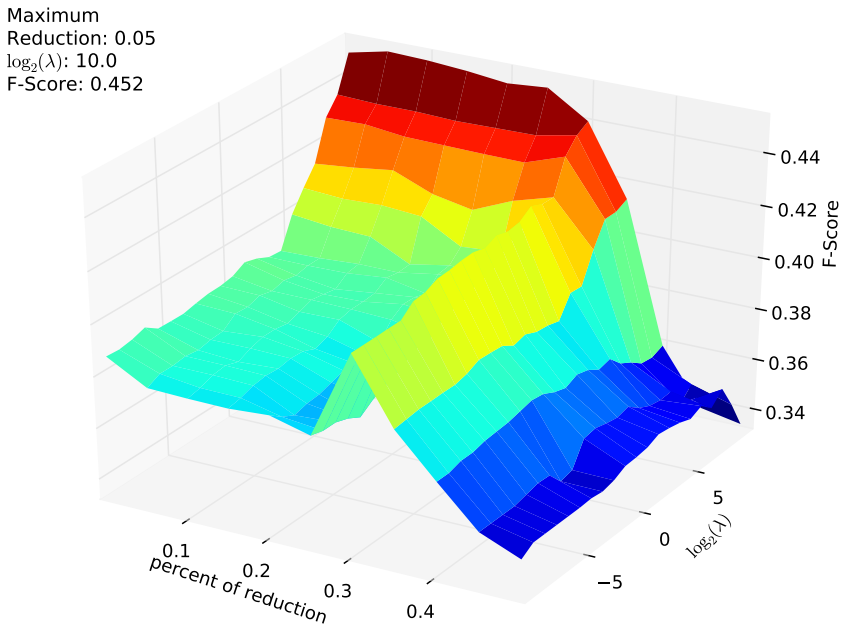

Figure 4.32: Maximum entropy 6-fold averaging for orthogonal sparse bigrams over tiles

4.4 Support Vector Machine Parameter Tuning

As discussed in Chapter 2, SVM has two parameters, the cost, C, and the kernel parameter, γ. C is a penalty for misclassification, and γ controls the linearity of the hyperplane. Lower values of γ force the hyperplane to be more linear, while higher values allow for closer fitting of the hyperplane to the data. The results of the parameter tuning experiments show that a high C increased F-score for posts, while a lower C increased the F-score for tiles. These values may be the result of the class labeling process described in section 3.4. Tile were labeled persuasive if a single post was labeled as persuasive. This means that it is possible for a long tile to contain a low proportion of persuasive posts. The result is a tile with many of the same features as a non-persuasive tile, but a different class label. The maximum F-scores for posts experiments were achieved by low γ values. Tile experiments performed best with slightly higher γ values than posts, but still less than 1. No parameter tuning was conducted over the percentage of feature reduction. As a result, all SVM experiments used every feature in the feature set.

Table 4.4 shows the parameters used for all subsequent SVM experiments. Again, note that some parameter sets performed equally well during parameter tuning (see Figures 4.33 through 4.48). In these cases, the smallest γ value and highest C were chosen to discourage over-fitting.

Support Vector Machine				
	Posts		Tiles	
Features	C	γ	C	γ
Unigrams	2^{15}	2^{-15}	2^{-1}	2^{-7}
Bigrams	2^{7}	2^{-7}	2^{1}	2^{-7}
Gappy	2^{7}	2^{-11}	2^{1}	2^{-9}
OSBs	2^{13}	2^{-11}	2^{3}	2^{-9}

Table 4.4: SVM parameters

4.4.1 Support Vector Machine over Posts

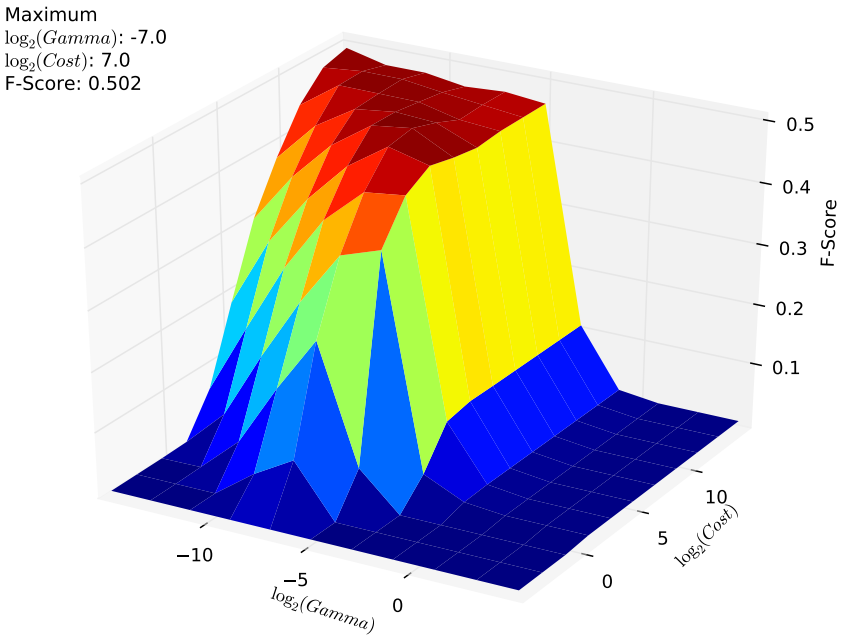

Figure 4.33: SVM 10-fold averaging for unigrams over posts

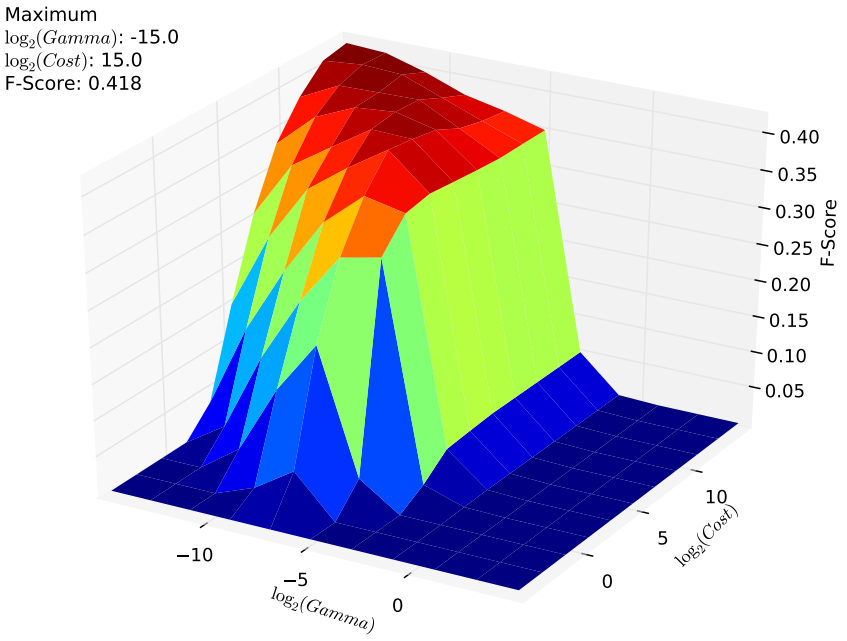

Figure 4.34: SVM 6-fold averaging for unigrams over posts

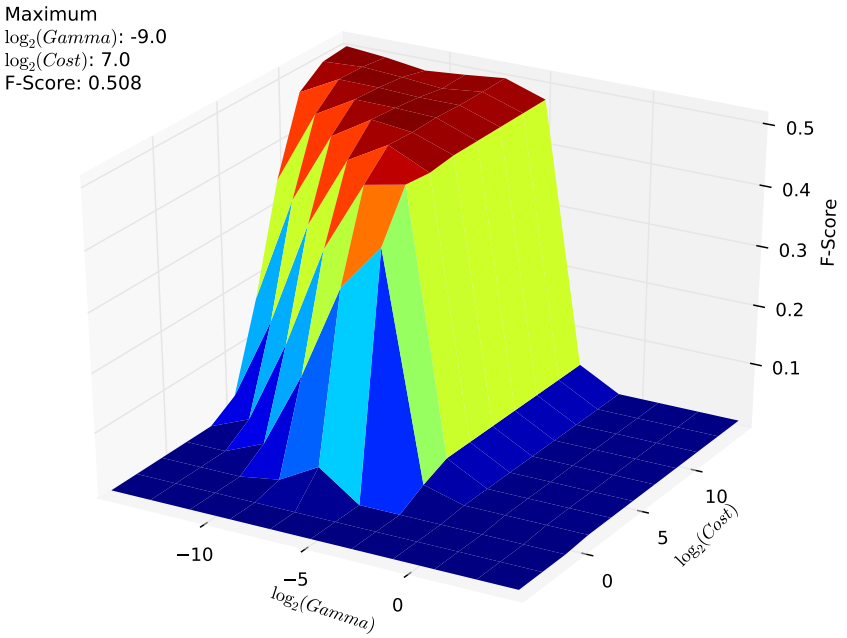

Figure 4.35: SVM 10-fold averaging for bigrams over posts

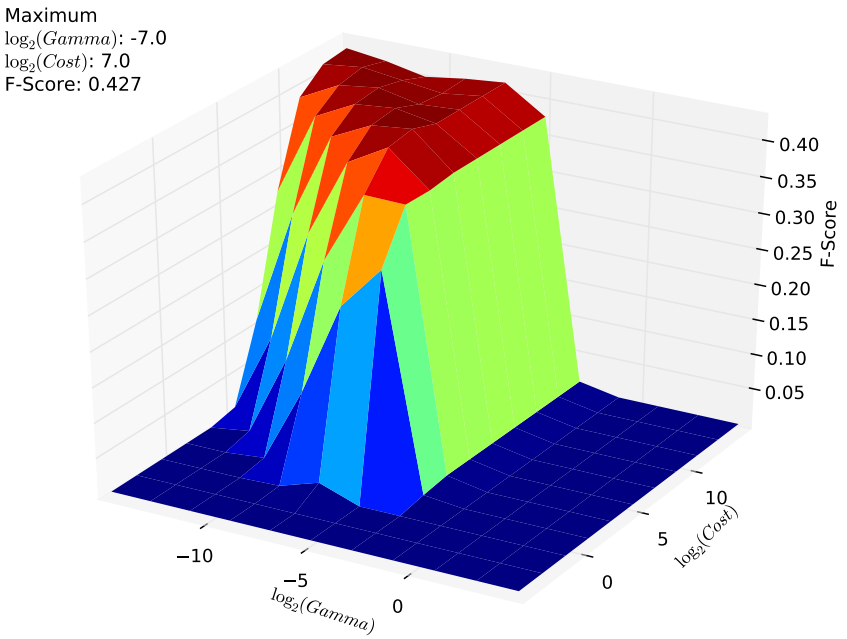

Figure 4.36: SVM 6-fold averaging for bigrams over posts

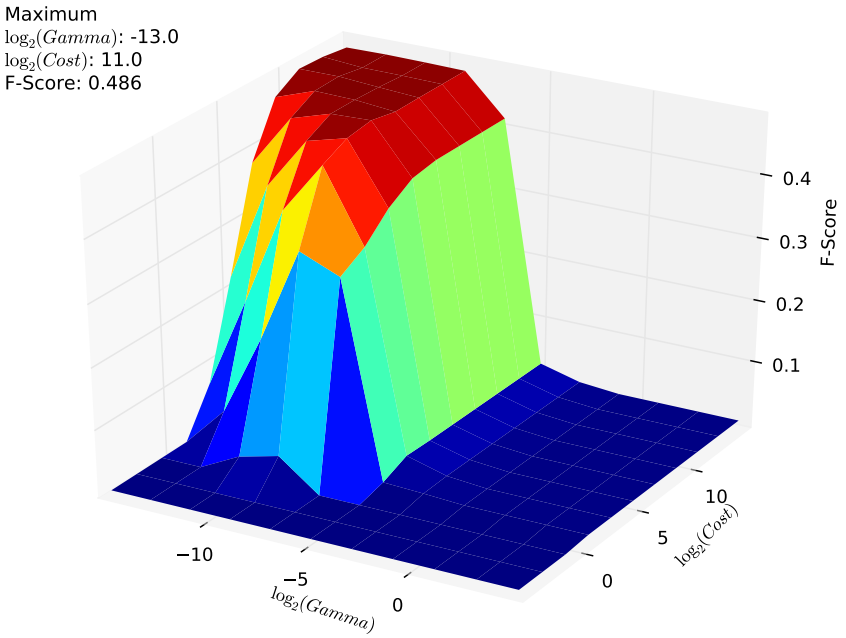

Figure 4.37: SVM 10-fold averaging for gappy bigrams over posts

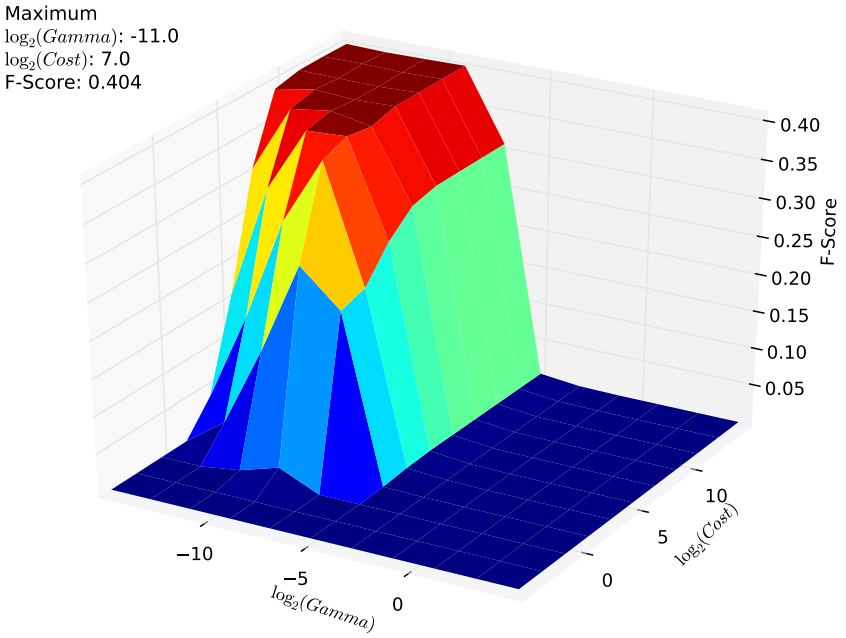

Figure 4.38: SVM 6-fold averaging for gappy bigrams over posts

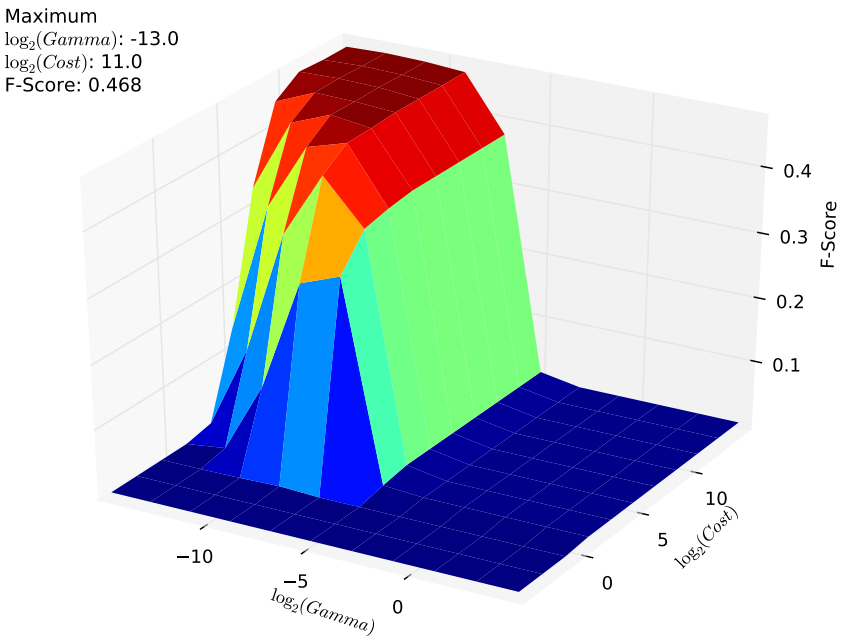

Figure 4.39: SVM 10-fold averaging for orthogonal sparse bigrams over posts

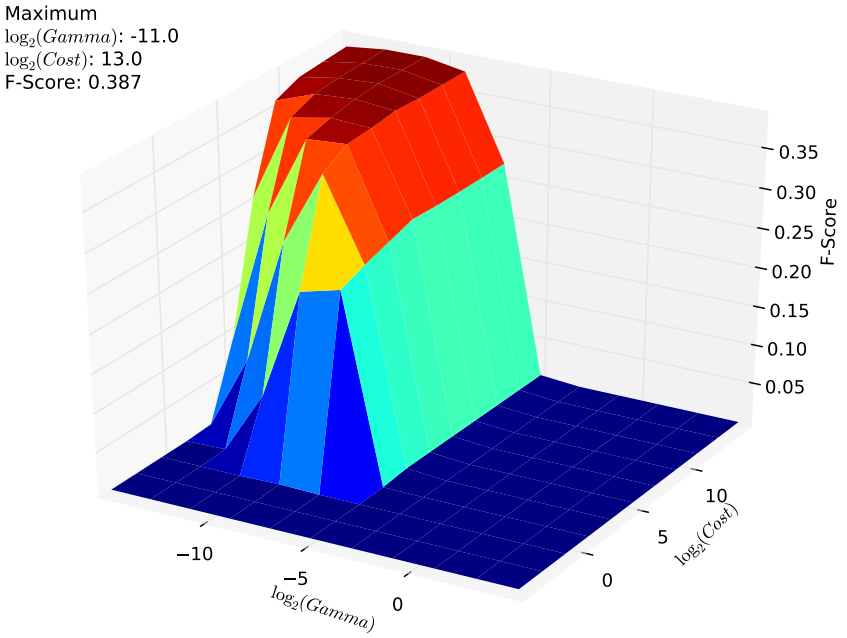

Figure 4.40: SVM 6-fold averaging for orthogonal sparse bigrams over posts

4.4.2 Support Vector Machine over Tiles

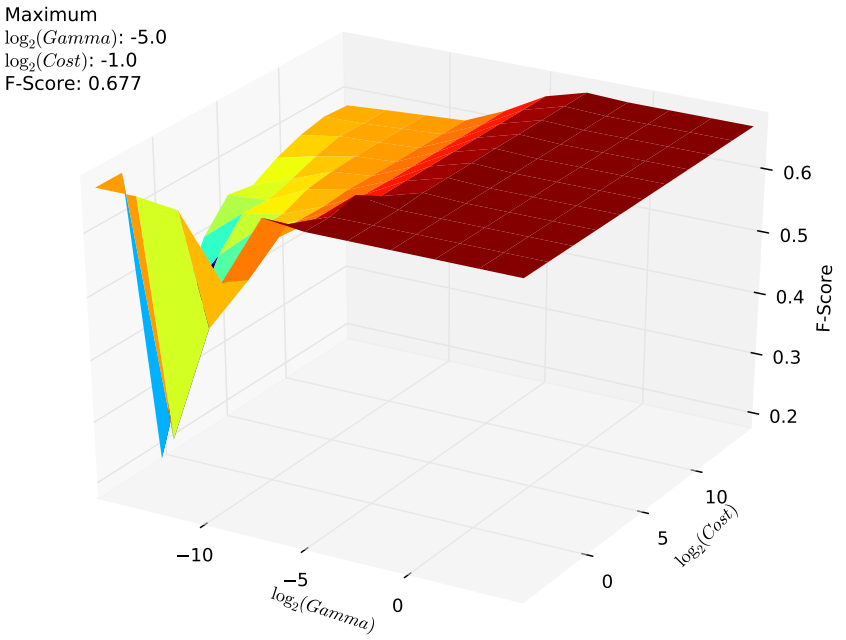

Figure 4.41: SVM 10-fold averaging for unigrams over tiles

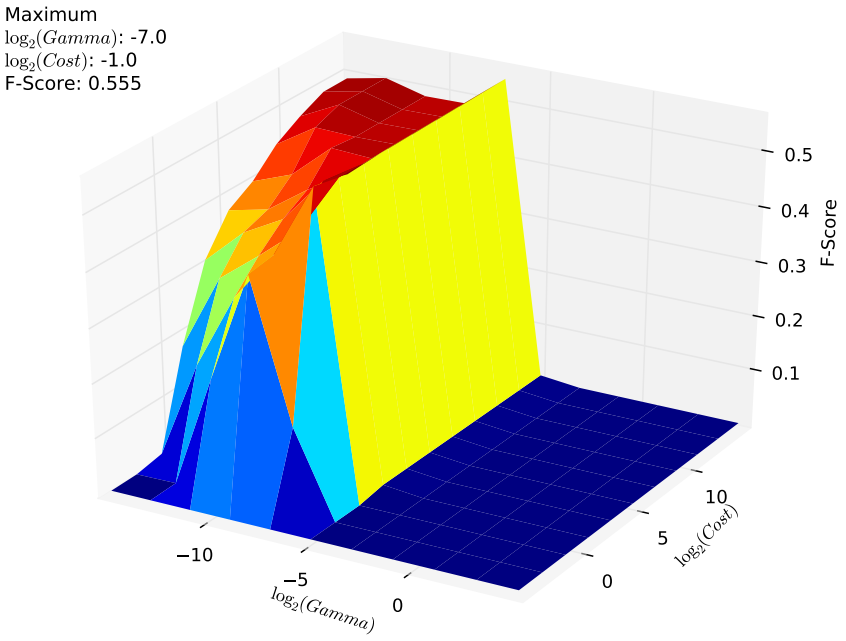

Figure 4.42: SVM 6-fold averaging for unigrams over tiles

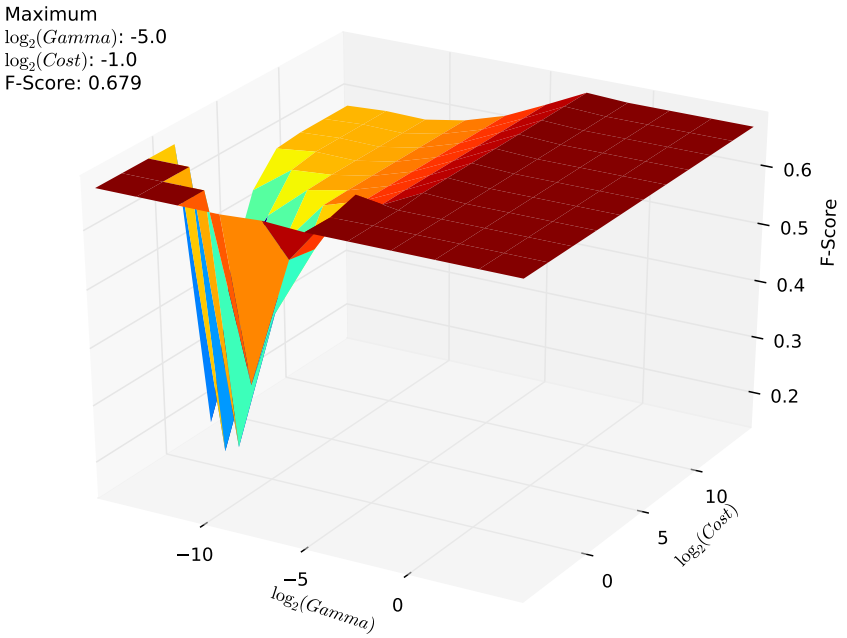

Figure 4.43: SVM 10-fold averaging for bigrams over tiles

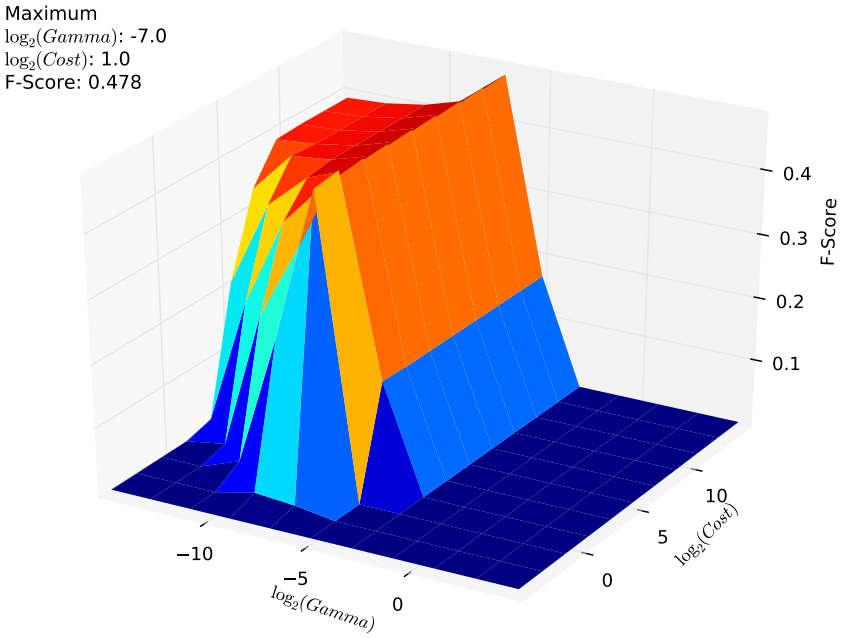

Figure 4.44: SVM 6-fold averaging for bigrams over tiles

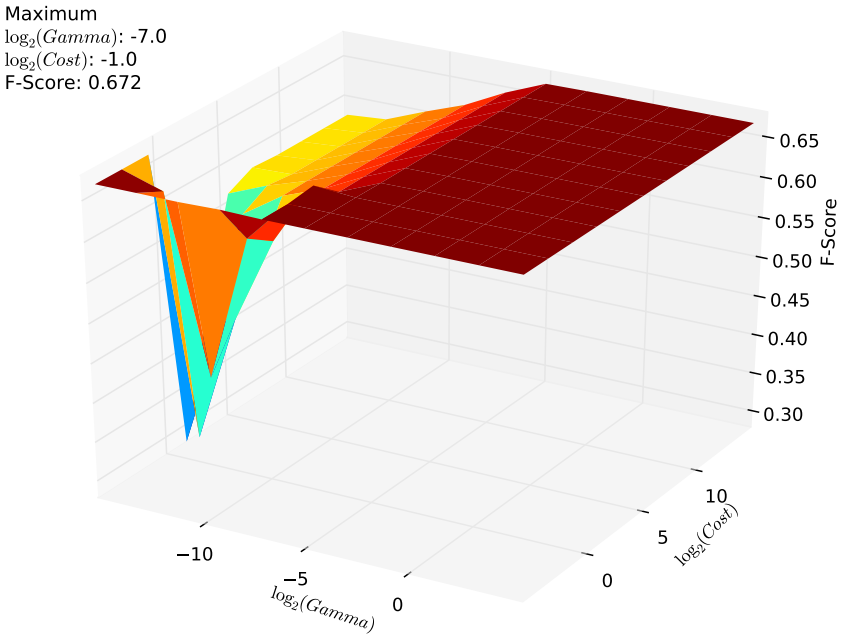

Figure 4.45: SVM 10-fold averaging for gappy bigrams over tiles

tiles

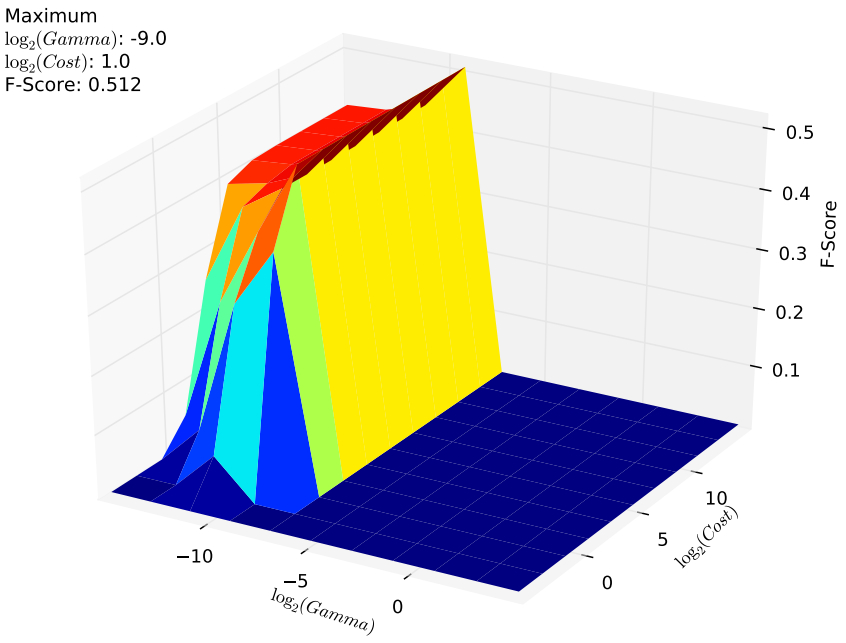

Figure 4.46: SVM 6-fold averaging for gappy bigrams over tiles

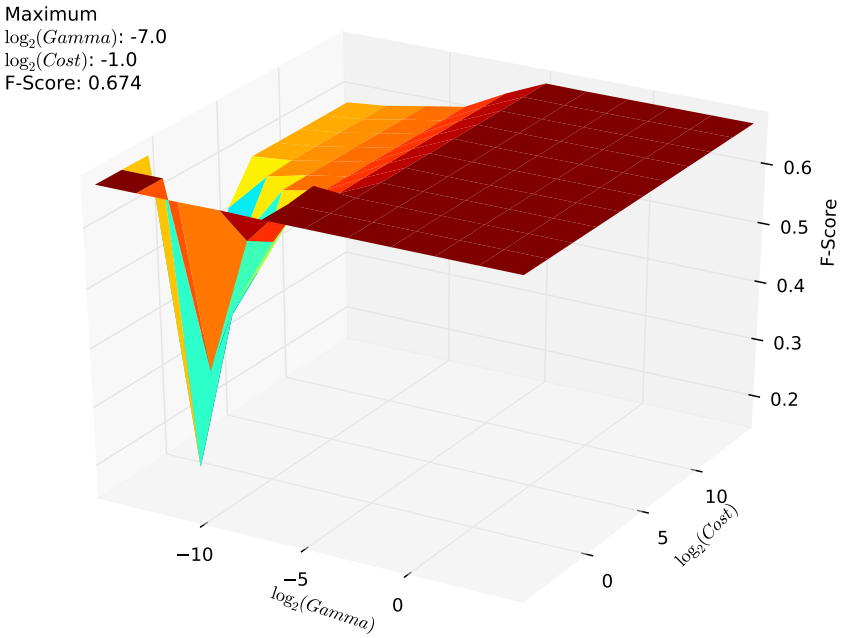

Figure 4.47: SVM 10-fold averaging for orthogonal sparse bigrams over tiles

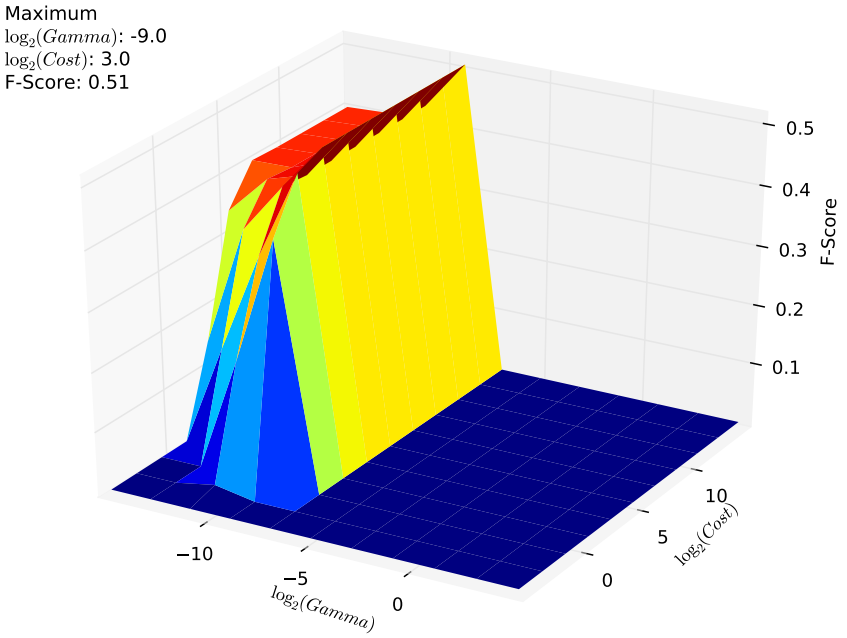

Figure 4.48: SVM 6-fold averaging for orthogonal sparse bigrams over tiles

4.5 Effects of Randomization Scheme on Parameter Tuning

As a result of the parameter tuning process, we gained valuable insight into the effect of different approaches to randomization. As described in section 3.5 and section 3.6, there were two approaches used to find parameters. The results of both sets of experiments are included in Figures 4.1 through 4.48. When comparing the two methods, it was evident that the repetition, noted in section 3.6, affected the results of the parameter tuning experiments for both posts and tiles. For this reason, the peaks of the first method are higher than the peaks of the second method for all metrics. The parameter shown in Table 4.2, Table 4.3, and Table 4.4 are the result of this second approach to randomization. Interestingly, the placement of the peaks in both sets of results are similar for post, but not for tiles. This lends confidence to our conclusion that the parameters in Table 4.2, Table 4.3, and Table 4.4 are appropriate for subsequent post experiment. The results for tiles vary greatly across the two sets of experiments. The cause of this disparity may be reflected in the subsequent results.

Having selected a set of parameters, the next task is to test the validity of these parameters over 5 repetitions of 6-fold validation, as well as to analyze trends in the results. The first set of results will show the performance of each technique over posts using four feature sets.

4.6 Six-fold Cross-validation with 5 Repetitions over Posts

The results of the experiments described in section 3.6 using posts are presented in this section. For this set of results and all subsequent sets of results, the baseline accuracy is the accuracy that would result from classifying all the posts or tiles in the test set as the most common class in the training set. The baseline F-score is calculated is the F-score that would result from classifying all posts or tiles as persuasive. The percent change is calculated as a percentage of the baseline metric.

4.6.1 Maximum Entropy

Table 4.5 shows that accuracy across all feature types was slightly better than the baseline accuracy, when using maximum entropy. F-scores for experiments with unigrams were the highest. In comparison to unigrams, experiments with bigrams had increased precision, but decreased recall. Gappy bigrams experiments showed lower precision and recall than unigram experiments. OSB experiments had the highest precision but the lowest recall, with the exception of repetition 4. The general trend is an exchange of increased precision at the expense of decreased recall, with the gappy bigram experiments as the exception.

Repetition 1								
Features	Accuracy	Baseline Accuracy	% Change	Precision	Recall	F-Score	Baseline F-score	% Change
unigrams	0.894	0.887	0.8	0.53	0.368	0.432	0.202	113.9
bigrams	0.9	0.887	1.5	0.596	0.298	0.394	0.202	95.0
GBGs	0.891	0.887	0.5	0.519	0.307	0.384	0.202	90.1
OSBs	0.899	0.887	1.4	0.608	0.257	0.36	0.202	78.2
Repetition 2								
Features	Accuracy	Baseline Accuracy	% Change	Precision	Recall	F-Score	Baseline F-score	% Change
unigrams	0.895	0.886	1.0	0.548	0.38	0.446	0.203	119.7
bigrams	0.898	0.886	1.4	0.596	0.308	0.402	0.203	98.0
GBGs	0.89	0.886	0.5	0.516	0.305	0.382	0.203	88.2
OSBs	0.897	0.886	1.2	0.605	0.255	0.355	0.203	74.9
Repetition 3								
Features	Accuracy	Baseline Accuracy	% Change	Precision	Recall	F-Score	Baseline F-score	% Change
unigrams	0.896	0.888	0.9	0.539	0.371	0.435	0.201	116.4
bigrams	0.9	0.888	1.4	0.601	0.313	0.404	0.201	101.0
GBGs	0.895	0.888	0.8	0.544	0.32	0.399	0.201	98.5
OSBs	0.898	0.888	1.1	0.592	0.26	0.353	0.201	75.6
Repetition 4								
Features	Accuracy	Baseline Accuracy	% Change	Precision	Recall	F-Score	Baseline F-score	% Change
unigrams	0.896	0.889	0.8	0.537	0.354	0.425	0.199	113.6
bigrams	0.898	0.889	1.0	0.574	0.283	0.376	0.199	88.9
GBGs	0.891	0.889	0.2	0.511	0.31	0.383	0.199	92.5
OSBs	0.898	0.889	1.0	0.587	0.249	0.345	0.199	73.4
Repetition 5								
Features	Accuracy	Baseline Accuracy	% Change	Precision	Recall	F-Score	Baseline F-score	% Change
unigrams	0.896	0.886	1.1	0.561	0.377	0.45	0.204	120.6
bigrams	0.899	0.886	1.5	0.608	0.304	0.404	0.204	98.0
GBGs	0.892	0.886	0.7	0.55	0.321	0.403	0.204	97.5
OSBs	0.899	0.886	1.5	0.629	0.261	0.368	0.204	80.4

Table 4.5: Maximum entropy over posts

The different performance among the three types of bigram experiments shows a few interesting changes. Experiments using gappy bigrams have decreased precision, while recall remains the same or slightly increases. Experiments using OSBs have a decreased recall with increased or unchanged precision. Overall, modified versions of traditional bigrams do not help maximum entropy for this classification task. This may be due to combining words that are not dependent on each other to form features. However, gappy bigrams and OSBs may be useful for other methods that are discussed later in this section.

4.6.2 Naive Bayes

The results of the naive bayes experiments, contained in Table 4.6, show accuracy was slightly better than the baseline accuracy across all feature types, except bigrams. F-scores for experiments with gappy bigrams were the highest. As compared to unigrams, experiments with

Repetition 1								
Features	Accuracy	Baseline Accuracy	% Change	Precision	Recall	F-Score	Baseline F-score	% Change
unigrams	0.887	0.887	0.0	0.495	0.401	0.438	0.202	116.8
bigrams	0.881	0.887	-0.7	0.464	0.419	0.437	0.202	116.3
GBGs	0.888	0.887	0.1	0.5	0.462	0.475	0.202	135.1
OSBs	0.894	0.887	0.8	0.537	0.385	0.444	0.202	119.8
Repetition 2								
Features	Accuracy	Baseline Accuracy	% Change	Precision	Recall	F-Score	Baseline F-score	% Change
unigrams	0.888	0.886	0.2	0.504	0.409	0.447	0.203	120.2
bigrams	0.879	0.886	-0.8	0.464	0.431	0.44	0.203	116.7
GBGs	0.89	0.886	0.5	0.516	0.459	0.479	0.203	136.0
OSBs	0.892	0.886	0.7	0.542	0.374	0.433	0.203	113.3
Repetition 3								
Features	Accuracy	Baseline Accuracy	% Change	Precision	Recall	F-Score	Baseline F-score	% Change
unigrams	0.889	0.888	0.1	0.499	0.402	0.443	0.201	120.4
bigrams	0.877	0.888	-1.2	0.441	0.412	0.424	0.201	110.9
GBGs	0.889	0.888	0.1	0.498	0.463	0.476	0.201	136.8
OSBs	0.892	0.888	0.5	0.522	0.367	0.428	0.201	112.9
Repetition 4								
Features	Accuracy	Baseline Accuracy	% Change	Precision	Recall	F-Score	Baseline F-score	% Change
unigrams	0.889	0.889	0.0	0.489	0.395	0.437	0.199	119.6
bigrams	0.879	0.889	-1.1	0.446	0.405	0.423	0.199	112.6
GBGs	0.891	0.889	0.2	0.504	0.438	0.467	0.199	134.7
OSBs	0.895	0.889	0.7	0.541	0.354	0.425	0.199	113.6
Repetition 5								
Features	Accuracy	Baseline Accuracy	% Change	Precision	Recall	F-Score	Baseline F-score	% Change
unigrams	0.889	0.886	0.3	0.511	0.419	0.46	0.204	125.5
bigrams	0.88	0.886	-0.7	0.471	0.419	0.443	0.204	117.2
GBGs	0.89	0.886	0.5	0.518	0.464	0.489	0.204	139.7
OSBs	0.895	0.886	1.0	0.559	0.376	0.449	0.204	120.1

Table 4.6: Naive bayes over posts

bigrams had lower precision, but increased recall. Gappy bigrams experiments showed higher precision and recall than unigram experiments. OSB experiments had the highest precision but the lowest recall.

The performance across the three types of bigrams shows a few interesting differences. Experiments using gappy bigrams have increased precision and recall. Experiments using OSBs have increased recall with decreased precision. Overall, gappy bigrams performed the best of all feature sets, while OSBs had a similar performance to traditional bigrams. This may be due to the naive bayes assumption that the occurrence of features is independent. Since the gappy bigrams and OSBs were constructed using and inter-word difference of 4 or less, this assumption is nearer to the truth. This explains the increased precision of both sets of modified bigrams. The inter-word distance also plays a role in the recall of the modified bigrams. Gappy bigrams

collapse OSBs with the same word pair into into a single feature, ignoring the inter-word distance (illustrated in Figure 2.1 and Figure 2.2). The preservation of the inter-word distance is what allows OSBs classify posts more precisely, but at the cost of recall.

4.6.3 Support Vector Machine

Repetition 1								
Features	Accuracy	Baseline Accuracy	% Change	Precision	Recall	F-Score	Baseline F-score	% Change
unigrams	0.888	0.887	0.1	0.493	0.361	0.414	0.202	105.0
bigrams	0.891	0.887	0.5	0.517	0.341	0.408	0.202	102.0
GBGs	0.895	0.887	0.9	0.547	0.313	0.395	0.202	95.5
OSBs	0.895	0.887	0.9	0.543	0.292	0.377	0.202	86.6
Repetition 2								
Features	Accuracy	Baseline Accuracy	% Change	Precision	Recall	F-Score	Baseline F-score	% Change
unigrams	0.887	0.886	0.1	0.506	0.365	0.42	0.203	106.9
bigrams	0.89	0.886	0.5	0.523	0.344	0.412	0.203	103.0
GBGs	0.894	0.886	0.9	0.552	0.309	0.395	0.203	94.6
OSBs	0.894	0.886	0.9	0.561	0.292	0.381	0.203	87.7
Repetition 3								
Features	Accuracy	Baseline Accuracy	% Change	Precision	Recall	F-Score	Baseline F-score	% Change
unigrams	0.889	0.888	0.1	0.497	0.361	0.414	0.201	106.0
bigrams	0.892	0.888	0.5	0.518	0.35	0.411	0.201	104.5
GBGs	0.894	0.888	0.7	0.541	0.325	0.397	0.201	97.5
OSBs	0.894	0.888	0.7	0.545	0.305	0.382	0.201	90.0
Repetition 4								
Features	Accuracy	Baseline Accuracy	% Change	Precision	Recall	F-Score	Baseline F-score	% Change
unigrams	0.889	0.889	0.0	0.487	0.346	0.404	0.199	103.0
bigrams	0.89	0.889	0.1	0.501	0.326	0.393	0.199	97.5
GBGs	0.895	0.889	0.7	0.535	0.302	0.382	0.199	92.0
OSBs	0.893	0.889	0.4	0.53	0.276	0.358	0.199	79.9
Repetition 5								
Features	Accuracy	Baseline Accuracy	% Change	Precision	Recall	F-Score	Baseline F-score	% Change
unigrams	0.886	0.886	0.0	0.501	0.358	0.415	0.204	103.4
bigrams	0.892	0.886	0.7	0.535	0.342	0.416	0.204	103.9
GBGs	0.897	0.886	1.2	0.583	0.322	0.414	0.204	102.9
OSBs	0.895	0.886	1.0	0.57	0.295	0.387	0.204	89.7

Table 4.7: Support vector machine over posts

The results of the SVM experiments are shown in Table 4.7 Again, these results show that accuracy was slightly better than the baseline accuracy across all feature types, with two exceptions (unigrams in repetitions 4 and 5). F-scores for experiments with unigrams were the highest (excepting repetition 5). As compared to unigrams, experiments with all three types of bigrams had higher precision and lower recall. This decrease in recall caused lower F-scores for all three types of bigrams.

The different performance among the three types of bigram experiments shows a few interesting

trends. Experiments using gappy bigrams and OSBs had higher precision and lower recall than bigrams. As in the the maximum entropy experiments, this may be due to combining words that are not dependent on each other to form features. Overall, modified versions of traditional bigrams do not help SVM for this classification task, as unigram and traditional bigram experiments had higher F-scores.

4.6.4 Overall Feature Performance

Using unigrams as features produced the highest F-scores for both maximum entropy and support vector machine experiments. While gappy bigrams produced the highest F-scores for naive bayes. Since maximum entropy and support vector machines are discriminative approaches, this is not an unexpected result. Generative models indicate which class is more likely, while discriminative models indicate which class is most similar. OSBs and gappy bigrams may introduce information that distorts the similarity of the two classes. Surprisingly, all three techniques resulted in low recall and high precision using OSBs as features. Based on the success of Cormack et al. when classifying SMS messages, blog comments, and emails summary information [9], it was expected that OSBs would perform much better. However, these results suggest that OSBs have discriminative power and may be used to cascade a high recall classifier with a high precision classifier.

4.7 Six-fold Cross-validation with 5 Repetitions over Tiles

The results of the experiments described in section 3.6 using tiles are presented in this section.

4.7.1 Maximum Entropy

Table 4.8 shows the results of the maximum entropy experiments. Accuracy across all three types of bigrams was slightly better than the baseline accuracy. Unigrams resulted in accuracies below the baseline with the exception of repetition 5. F-scores for experiments using unigrams were highest. Experiments with all bigram types had increased precision, but decreased recall. Gappy bigram experiments showed lower precision and recall than unigram experiments. The F-scores in this set of experiments suggests that tiles are not useful for this method of machine learning (see section 4.7.4 for further discussion).

4.7.2 Naive Bayes

The results of the naive bayes experiments are contained in Table 4.9. Accuracy was better than the baseline accuracy for gappy bigram experiments. Unigram, bigram and OSB experiments all produced accuracies below the baseline accuracies (excepting OSBs in repetitions 3 and

Repetition 1								
Features	Accuracy	Baseline Accuracy	% Change	Precision	Recall	F-Score	Baseline F-score	% Change
unigrams	0.579	0.603	-4.0	0.504	0.502	0.488	0.575	-15.1
bigrams	0.62	0.603	2.8	0.549	0.332	0.399	0.575	-30.6
GBGs	0.627	0.603	4.0	0.558	0.41	0.453	0.575	-21.2
OSBs	0.615	0.603	2.0	0.525	0.373	0.418	0.575	-27.3
Repetition 2								
Features	Accuracy	Baseline Accuracy	% Change	Precision	Recall	F-Score	Baseline F-score	% Change
unigrams	0.572	0.587	-2.6	0.497	0.526	0.505	0.592	-14.7
bigrams	0.611	0.587	4.1	0.556	0.335	0.399	0.592	-32.6
GBGs	0.621	0.587	5.8	0.556	0.426	0.466	0.592	-21.3
OSBs	0.617	0.587	5.1	0.554	0.383	0.433	0.592	-26.9
Repetition 3								
Features	Accuracy	Baseline Accuracy	% Change	Precision	Recall	F-Score	Baseline F-score	% Change
unigrams	0.584	0.577	1.2	0.495	0.5	0.492	0.593	-17.0
bigrams	0.612	0.577	6.1	0.565	0.323	0.41	0.593	-30.9
GBGs	0.641	0.577	11.1	0.608	0.43	0.492	0.593	-17.0
OSBs	0.621	0.577	7.6	0.566	0.361	0.437	0.593	-26.3
Repetition 4								
Features	Accuracy	Baseline Accuracy	% Change	Precision	Recall	F-Score	Baseline F-score	% Change
unigrams	0.571	0.591	-3.4	0.475	0.494	0.479	0.588	-18.5
bigrams	0.614	0.591	3.9	0.569	0.312	0.4	0.588	-32.0
GBGs	0.639	0.591	8.1	0.606	0.406	0.479	0.588	-18.5
OSBs	0.622	0.591	5.2	0.573	0.363	0.441	0.588	-25.0
Repetition 5								
Features	Accuracy	Baseline Accuracy	% Change	Precision	Recall	F-Score	Baseline F-score	% Change
unigrams	0.578	0.576	0.3	0.521	0.515	0.517	0.612	-15.5
bigrams	0.599	0.576	4.0	0.609	0.358	0.432	0.612	-29.4
GBGs	0.6	0.576	4.2	0.576	0.444	0.488	0.612	-20.3
OSBs	0.592	0.576	2.8	0.582	0.405	0.452	0.612	-26.1

Table 4.8: Maximum entropy over tiles

4). Gappy bigrams produced the highest precision, while unigrams produced the lowest. Conversely, unigrams had the highest recall, while gappy bigrams had the lowest. The overall trend was increased precision with a decrease in recall. Due to the disparity is the precision-recall trade off, there is no trend in F-score. All F-score were below the baseline F-score. This set of results along with the results presented in section 4.7.1 suggest that tiles may not be usable for any of our machine learning techniques (see section 4.7.4 for further discussion).

4.7.3 Support Vector Machine

Table 4.10 shows the results of the SVM experiments. Accuracy was better than the baseline accuracy across all feature types in repetitions 3, 4 and 5. F-scores for experiments using gappy bigrams were highest, with the exception of repetition 3. In comparison to unigram experiments, all three types of bigram experiments generally had higher precision, but lower recall. This third

Repetition 1								
Features	Accuracy	Baseline Accuracy	% Change	Precision	Recall	F-Score	Baseline F-score	% Change
unigrams	0.487	0.603	-19.2	0.415	0.728	0.526	0.576	-8.7
bigrams	0.52	0.603	-13.8	0.433	0.687	0.526	0.575	-8.5
GBGs	0.612	0.603	1.5	0.497	0.57	0.52	0.575	-9.6
OSBs	0.563	0.603	-6.6	0.457	0.594	0.508	0.575	-11.7
Repetition 2								
Features	Accuracy	Baseline Accuracy	% Change	Precision	Recall	F-Score	Baseline F-score	% Change
unigrams	0.515	0.587	-12.3	0.445	0.737	0.553	0.592	-6.6
bigrams	0.543	0.587	-7.5	0.456	0.662	0.536	0.592	-9.5
GBGs	0.612	0.587	4.3	0.514	0.615	0.547	0.592	-7.6
OSBs	0.569	0.587	-3.1	0.482	0.622	0.537	0.592	-9.3
Repetition 3								
Features	Accuracy	Baseline Accuracy	% Change	Precision	Recall	F-Score	Baseline F-score	% Change
unigrams	0.518	0.577	-10.2	0.447	0.699	0.541	0.593	-8.8
bigrams	0.552	0.577	-4.3	0.466	0.626	0.529	0.593	-10.8
GBGs	0.619	0.577	7.3	0.537	0.584	0.547	0.593	-7.8
OSBs	0.581	0.577	0.7	0.491	0.598	0.535	0.593	-9.8
Repetition 4								
Features	Accuracy	Baseline Accuracy	% Change	Precision	Recall	F-Score	Baseline F-score	% Change
unigrams	0.517	0.591	-12.5	0.445	0.709	0.546	0.588	-7.1
bigrams	0.548	0.591	-7.3	0.463	0.636	0.534	0.588	-9.2
GBGs	0.619	0.591	4.7	0.532	0.571	0.544	0.588	-7.5
OSBs	0.593	0.591	0.3	0.499	0.611	0.548	0.588	-6.8
Repetition 5								
Features	Accuracy	Baseline Accuracy	% Change	Precision	Recall	F-Score	Baseline F-score	% Change
unigrams	0.507	0.576	-12.0	0.458	0.745	0.565	0.612	-7.7
bigrams	0.539	0.577	-6.6	0.484	0.704	0.571	0.612	-6.7
GBGs	0.605	0.577	4.9	0.548	0.631	0.58	0.612	-5.2
OSBs	0.56	0.577	-2.9	0.502	0.633	0.556	0.612	-9.2

Table 4.9: Naive bayes over tiles

set of results confirms that tiles do not allow any of our machine learning techniques to detect persuasion (see section 4.7.4 for further discussion).

4.7.4 Performance of TextTiling

Since none of the three machine learning techniques used in this research outperformed the baseline F-score, TextTiling did not provide a suitable method of segmentation for this classifications task. This is probably due to the fact that the criteria for labeling a post as persuasive was that it contain one persuasive post. Due to this labeling scheme described in section 3.4, there is a high probability that persuasive tiles had many of the same features as non-persuasive tiles. Future research should include an exploration of other segmentation techniques.

Repetition 1								
Features	Accuracy	Baseline Accuracy	% Change	Precision	Recall	F-Score	Baseline F-score	% Change
unigrams	0.573	0.603	-5.0	0.502	0.433	0.418	0.575	-27.3
bigrams	0.591	0.603	-2.0	0.493	0.397	0.435	0.575	-24.3
GBGs	0.609	0.603	1.0	0.525	0.443	0.473	0.575	-17.7
OSBs	0.599	0.603	-0.7	0.52	0.402	0.443	0.575	-23.0
Repetition 2								
Features	Accuracy	Baseline Accuracy	% Change	Precision	Recall	F-Score	Baseline F-score	% Change
unigrams	0.58	0.587	-1.2	0.525	0.471	0.45	0.592	-24.0
bigrams	0.598	0.587	1.9	0.537	0.423	0.466	0.592	-21.3
GBGs	0.606	0.587	3.2	0.546	0.457	0.491	0.592	-17.1
OSBs	0.601	0.587	2.4	0.55	0.416	0.462	0.592	-22.0
Repetition 3								
Features	Accuracy	Baseline Accuracy	% Change	Precision	Recall	F-Score	Baseline F-score	% Change
unigrams	0.612	0.577	6.1	0.543	0.559	0.542	0.593	-8.6
bigrams	0.601	0.577	4.2	0.537	0.437	0.479	0.593	-19.2
GBGs	0.619	0.577	7.3	0.561	0.468	0.507	0.593	-14.5
OSBs	0.613	0.577	6.2	0.561	0.439	0.487	0.593	-17.9
Repetition 4								
Features	Accuracy	Baseline Accuracy	% Change	Precision	Recall	F-Score	Baseline F-score	% Change
unigrams	0.596	0.591	0.8	0.533	0.501	0.468	0.588	-20.4
bigrams	0.61	0.591	3.2	0.543	0.443	0.482	0.588	-18.0
GBGs	0.621	0.591	5.1	0.56	0.476	0.508	0.588	-13.6
OSBs	0.614	0.591	3.9	0.556	0.431	0.478	0.588	-18.7
Repetition 5								
Features	Accuracy	Baseline Accuracy	% Change	Precision	Recall	F-Score	Baseline F-score	% Change
unigrams	0.576	0.576	0.0	0.539	0.473	0.459	0.612	-25.0
bigrams	0.587	0.576	1.9	0.557	0.419	0.47	0.612	-23.2
GBGs	0.604	0.576	4.9	0.574	0.47	0.51	0.612	-16.7
OSBs	0.603	0.576	4.7	0.587	0.439	0.491	0.612	-19.8

Table 4.10: Support vector machine over tiles

4.8 Leave-One-Out over Posts

The results presented in section 4.6 indicated that it was possible to train weak classifiers using posts. In order to validate this belief, it is necessary to review the results of the experiments conducted using the first method described in section 3.7.

4.8.1 Maximum Entropy

Trained with all, except Rogan (13608 posts, 89.7% not persuasive)								
Features	Accuracy	Baseline Accuracy	% Change	Precision	Recall	F-Score	Baseline F-score	% Change
unigrams	0.865	0.85	1.8	0.594	0.334	0.428	0.262	63.4
bigrams	0.873	0.85	2.7	0.701	0.271	0.39	0.262	48.9
GBGs	0.869	0.85	2.2	0.628	0.313	0.417	0.262	59.2
OSBs	0.872	0.85	2.6	0.734	0.235	0.356	0.262	35.9
Trained with all, except Taylor (11944 posts, 89.7% not persuasive)								
Features	Accuracy	Baseline Accuracy	% Change	Precision	Recall	F-Score	Baseline F-score	% Change
unigrams	0.876	0.872	0.5	0.524	0.365	0.43	0.227	89.4
bigrams	0.883	0.872	1.3	0.605	0.237	0.341	0.227	50.2
GBGs	0.877	0.872	0.6	0.539	0.266	0.356	0.227	56.8
OSBs	0.882	0.872	1.1	0.632	0.186	0.288	0.227	26.9
Trained with all, except SDPolice (18033 posts, 88.7% not persuasive)								
Features	Accuracy	Baseline Accuracy	% Change	Precision	Recall	F-Score	Baseline F-score	% Change
unigrams	0.899	0.826	8.8	0.798	0.56	0.658	0.297	121.5
bigrams	0.906	0.826	9.7	0.892	0.525	0.661	0.297	122.6
GBGs	0.884	0.826	7.0	0.797	0.447	0.573	0.297	92.9
OSBs	0.896	0.826	8.5	0.89	0.461	0.607	0.297	104.4
Trained with all, except Waco (12986 post, 86.0% not persuasive)								
Features	Accuracy	Baseline Accuracy	% Change	Precision	Recall	F-Score	Baseline F-score	% Change
unigrams	0.91	0.937	-2.9	0.293	0.296	0.294	0.119	147.1
bigrams	0.918	0.937	-2.0	0.339	0.306	0.322	0.119	170.6
GBGs	0.906	0.937	-3.3	0.286	0.317	0.301	0.119	152.9
OSBs	0.918	0.937	-2.0	0.324	0.266	0.292	0.119	145.4

Table 4.11: Maximum entropy over posts, trained on three of four transcript types

The results of the maximum entropy experiments in Table 4.11 show some important differences in the behavior of each transcript type. The experiments trained without Rogan and without Taylor produced similar results. Each set of features performed the same with regards to the rank order of the results for each metric. In these experiments, the accuracies and F-scores were higher than the baseline metrics. Both types of experiments showed stronger precision, than recall.

The experiments trained without using the San Diego Police transcript boasted the highest scores for each metric. The accuracies exceeded baseline accuracies. This change in performance can mainly be attributed to the small size of the transcript (824 posts, 4.4% of the corpus).

Training on all transcripts except Waco produced lower scores for precision, recall and F-score than the other three types of experiments. None of the experiments in this set outperformed the baseline accuracy, but they did outperform the baseline F-score. This set of results begins to suggest that the Waco transcripts are somehow different than the other three transcript types.

4.8.2 Naive Bayes

Trained with all, except Rogan (13608 posts, 89.7% not persuasive)								
Features	Accuracy	Baseline Accuracy	% Change	Precision	Recall	F-Score	Baseline F-score	% Change
unigrams	0.854	0.85	0.5	0.52	0.389	0.445	0.262	69.8
bigrams	0.851	0.85	0.1	0.508	0.417	0.458	0.262	74.8
GBGs	0.863	0.85	1.5	0.559	0.418	0.478	0.262	82.4
OSBs	0.858	0.85	0.9	0.553	0.305	0.393	0.262	50.0
Trained with all, except Taylor (11944 posts, 89.7% not persuasive)								
Features	Accuracy	Baseline Accuracy	% Change	Precision	Recall	F-Score	Baseline F-score	% Change
unigrams	0.862	0.872	-1.1	0.453	0.373	0.409	0.227	80.2
bigrams	0.854	0.872	-2.1	0.425	0.399	0.411	0.227	81.1
GBGs	0.869	0.872	-0.3	0.486	0.41	0.445	0.227	96.0
OSBs	0.872	0.872	0.0	0.502	0.348	0.411	0.227	81.1
Trained with all, except SDPolice (18033 posts, 88.7% not persuasive)								
Features	Accuracy	Baseline Accuracy	% Change	Precision	Recall	F-Score	Baseline F-score	% Change
unigrams	0.888	0.826	7.5	0.731	0.56	0.635	0.297	113.8
bigrams	0.885	0.826	7.1	0.773	0.482	0.594	0.297	100.0
GBGs	0.892	0.826	8.0	0.737	0.596	0.659	0.297	121.9
OSBs	0.89	0.826	7.7	0.851	0.447	0.586	0.297	97.3
Trained with all, except Waco (12986 post, 86.0% not persuasive)								
Features	Accuracy	Baseline Accuracy	% Change	Precision	Recall	F-Score	Baseline F-score	% Change
unigrams	0.889	0.937	-5.1	0.27	0.444	0.335	0.119	181.5
bigrams	0.865	0.937	-7.7	0.242	0.53	0.332	0.119	179.0
GBGs	0.866	0.937	-7.6	0.244	0.532	0.334	0.119	180.7
OSBs	0.879	0.937	-6.2	0.258	0.481	0.336	0.119	182.4

Table 4.12: Naive bayes over posts, trained on three of four transcript types

Table 4.12 shows the results of the naive bayes experiments, which highlights some differences between each transcript type. The experiments trained without Rogan and without Taylor produced similar results. Each set of features performed the same with regards to the rank order of the results for each metric, except F-score. With regard to F-score, these two sets of experiments both show gappy bigrams producing the highest F-scores and bigrams the second highest. In these experiments, the Rogan experiment accuracies exceeded the baseline, while the Taylor experiment accuracies did not. Both sets had F-scores that were higher than the baseline F-scores. Both types of experiments showed stronger precision, than recall.

The experiments trained without using the San Diego Police transcript boasted the highest scores for each metric. The accuracies exceeded the baseline accuracies for all experiments.

Again, this change in performance can mainly be attributed to the small size of the transcript (824 posts, 4.4% of the corpus).

Training on all transcripts except Waco produced lower scores for precision and F-score than the other three types of experiments. All experiments in this set had lower accuracies than the baseline accuracies, but F-scores higher than the baseline F-score. One interesting characteristic of this set of results is the similarity of the F-scores. This occurred due to similar precision across all feature sets with minimal change to recall (less than 5%). This set results also suggests that Waco is different from the other transcript types. If Waco had been similar, we could have expected more variation across the results from the different feature sets.

4.8.3 Support Vector Machines

Trained with all, except Rogan (13608 posts, 89.7% not persuasive)								
Features	Accuracy	Baseline Accuracy	% Change	Precision	Recall	F-Score	Baseline F-score	% Change
unigrams	0.856	0.85	0.7	0.535	0.319	0.4	0.262	52.7
bigrams	0.87	0.85	2.4	0.631	0.325	0.429	0.262	63.7
GBGs	0.873	0.85	2.7	0.67	0.3	0.414	0.262	58.0
OSBs	0.872	0.85	2.6	0.681	0.285	0.401	0.262	53.1
Trained with all, except Taylor (11944 posts, 89.7% not persuasive)								
Features	Accuracy	Baseline Accuracy	% Change	Precision	Recall	F-Score	Baseline F-score	% Change
unigrams	0.87	0.872	-0.2	0.491	0.384	0.431	0.227	89.9
bigrams	0.875	0.872	0.3	0.527	0.269	0.356	0.227	56.8
GBGs	0.882	0.872	1.1	0.587	0.275	0.374	0.227	64.8
OSBs	0.882	0.872	1.1	0.602	0.241	0.344	0.227	51.5
Trained with all, except SDPolice (18033 posts, 88.7% not persuasive)								
Features	Accuracy	Baseline Accuracy	% Change	Precision	Recall	F-Score	Baseline F-score	% Change
unigrams	0.884	0.826	7.0	0.77	0.475	0.588	0.297	98.0
bigrams	0.904	0.826	9.4	0.818	0.574	0.675	0.297	127.3
GBGs	0.899	0.826	8.8	0.839	0.518	0.64	0.297	115.5
OSBs	0.902	0.826	9.2	0.897	0.496	0.639	0.297	115.2
Trained with all, except Waco (12986 post, 86.0% not persuasive)								
Features	Accuracy	Baseline Accuracy	% Change	Precision	Recall	F-Score	Baseline F-score	% Change
unigrams	0.906	0.937	-3.3	0.271	0.285	0.278	0.119	133.6
bigrams	0.904	0.937	-3.5	0.283	0.341	0.31	0.119	160.5
GBGs	0.903	0.937	-3.6	0.264	0.293	0.278	0.119	133.6
OSBs	0.909	0.937	-3.0	0.287	0.296	0.291	0.119	144.5

Table 4.13: Support vector machine over posts, trained on three of four transcript types

SVM produced the results in Table 4.13. The table shows important differences in the transcript types. This set of experiments continued the trend than experiments trained by leaving out San Diego Police performed the best, followed by leaving out Rogan. Leaving out Waco produces the worst precision, recall and F-score. It is important to note that similarities noted in section 4.8.1 and section 4.8.2 for the Rogan and Taylor experiments are not present in this

set of results. This set of result reinforce the claim that the Waco transcripts are different than the other three types.

4.8.4 Effects of Leave-One-Out for Posts

The results presented in this section suggest that the Rogan transcripts and the Taylor transcripts are similar with regard to post-level persuasion. Due to the difference in performance from the other three types of experiments, there is reason to believe that the Waco transcripts are significantly different at the post level than the other three types of transcripts. In the next section, we discuss the results for the same experiments using tiles.

4.9 Leave-One-Out over Tiles

The results presented in section 4.7 suggested that using tiles did not learn any signal that indicated persuasion. In order to validate this observation, it is necessary to review the results of the experiments conducted using the first method described in section 3.7.

4.9.1 Maximum Entropy

Trained with all, except Rogan (1195 tiles, 55.8% not persuasive)								
Features	Accuracy	Baseline Accuracy	% Change	Precision	Recall	F-Score	Baseline F-score	% Change
unigrams	0.591	0.516	14.5	0.61	0.429	0.503	0.652	-22.9
bigrams	0.541	0.516	4.8	0.542	0.329	0.41	0.652	-37.1
GBGs	0.578	0.516	12.0	0.59	0.417	0.488	0.652	-25.2
OSBs	0.557	0.516	7.9	0.561	0.381	0.454	0.652	-30.4
Trained with all, except Taylor (1052 tiles, 59.3% not persuasive)								
Features	Accuracy	Baseline Accuracy	% Change	Precision	Recall	F-Score	Baseline F-score	% Change
unigrams	0.542	0.53	2.3	0.603	0.401	0.481	0.693	-30.6
bigrams	0.515	0.53	-2.8	0.606	0.244	0.348	0.693	-49.8
GBGs	0.515	0.53	-2.8	0.57	0.347	0.431	0.693	-37.8
OSBs	0.515	0.53	-2.8	0.606	0.244	0.348	0.693	-49.8
Trained with all, except SDPolice (1638 tiles, 54.8% not persuasive)								
Features	Accuracy	Baseline Accuracy	% Change	Precision	Recall	F-Score	Baseline F-score	% Change
unigrams	0.628	0.513	22.4	0.634	0.65	0.642	0.678	-5.3
bigrams	0.603	0.513	17.5	0.68	0.425	0.523	0.678	-22.9
GBGs	0.577	0.513	12.5	0.64	0.4	0.492	0.678	-27.4
OSBs	0.615	0.513	19.9	0.75	0.375	0.5	0.678	-26.3
Trained with all, except Waco (1263 tiles, 49.0% not persuasive)								
Features	Accuracy	Baseline Accuracy	% Change	Precision	Recall	F-Score	Baseline F-score	% Change
unigrams	0.581	0.7	-17.0	0.341	0.426	0.379	0.462	-18.0
bigrams	0.521	0.7	-25.6	0.329	0.574	0.418	0.462	-9.5
GBGs	0.525	0.7	-25.0	0.333	0.581	0.424	0.462	-8.2
OSBs	0.446	0.7	-36.3	0.309	0.684	0.426	0.462	-7.8

Table 4.14: Maximum entropy over tiles, trained on three of four transcript types

Table 4.14 shows the results of maximum entropy over tiles. The experiments trained without Rogan and without Taylor produced similar results when using unigrams as features. The unigram experiments for these two sets produced the highest accuracy, recall, and F-score within each set. In the Rogan set, unigrams produced the highest precision. While in the Taylor set, unigrams resulted in the second highest precision, but within .003 of the highest precision. Interestingly, in the Taylor set of experiments, bigrams and OSBs have exactly the same results.

The experiments trained without using the San Diego Police transcript boasted the highest scores for each metric within each type of feature. The accuracies exceeded the the baseline accuracies for each set of features. This change in performance can mainly be attributed to the small size of the transcript(78 tiles, 4.5% of the corpus).

Within the Waco set, the lowest F-scores occur in the experiment using unigrams, while the highest occurs in the experiment using OSBs. This is opposite of the trends for the other sets of experiments. Since all F-scores are below the baseline F-scores, it is clear that tiles did not help maximum entropy to detect persuasion. Due to this fact it is unclear if any conclusion can be drawn about the different transcript types.

4.9.2 Naive Bayes

In the results of the experiments shown in Table 4.15, the experiments trained without Rogan unigrams resulted in the highest F-score and recall, while gappy bigrams resulted in the highest accuracy and precision. In the Taylor set, gappy bigrams produced the highest accuracy, precision, and F-score. Bigrams produced the highest recall.

In other sets of experiments, leaving out the San Diego transcript has dramatically, improved all metrics. However, this not the case in this experiments despite the small size of the transcript(78 tiles, 4.5% of the corpus). Additionally, in this set, gappy bigrams and OSBs resulted in the same accuracies, but different recall and precision. The Waco set generally produced the lowest accuracy, precision, and F-score for each feature set, but the highest recall for each feature set, except unigrams. Once again, all F-scores are below the baseline F-scores. Clearly, tiles did not help naive bayes to successfully detect persuasion. Due to this fact no conclusions can be drawn about the different transcript types from these results.

4.9.3 Support Vector Machine

The SVM experiment results are contained in Table 4.16. The experiments trained without Rogan using unigrams resulted in the highest scores across all metrics. Bigrams resulted in the

Trained with all, except Rogan (1195 tiles, 55.8% not persuasive)								
Features	Accuracy	Baseline Accuracy	% Change	Precision	Recall	F-Score	Baseline F-score	% Change
unigrams	0.522	0.516	1.2	0.503	0.869	0.638	0.652	-2.1
bigrams	0.52	0.516	0.8	0.502	0.845	0.63	0.652	-3.4
GBGs	0.568	0.516	10.1	0.538	0.754	0.628	0.652	-3.7
OSBs	0.547	0.516	6.0	0.522	0.758	0.618	0.652	-5.2
Trained with all, except Taylor (1052 tiles, 59.3% not persuasive)								
Features	Accuracy	Baseline Accuracy	% Change	Precision	Recall	F-Score	Baseline F-score	% Change
unigrams	0.473	0.53	-10.8	0.503	0.474	0.488	0.693	-29.6
bigrams	0.498	0.53	-6.0	0.528	0.514	0.521	0.693	-24.8
GBGs	0.565	0.53	6.6	0.609	0.5	0.549	0.693	-20.8
OSBs	0.529	0.53	-0.2	0.587	0.375	0.458	0.693	-33.9
Trained with all, except SDPolice (1638 tiles, 54.8% not persuasive)								
Features	Accuracy	Baseline Accuracy	% Change	Precision	Recall	F-Score	Baseline F-score	% Change
unigrams	0.526	0.513	2.5	0.523	0.85	0.648	0.678	-4.4
bigrams	0.5	0.513	-2.5	0.509	0.725	0.598	0.678	-11.8
GBGs	0.577	0.513	12.5	0.59	0.575	0.582	0.678	-14.2
OSBs	0.577	0.513	12.5	0.569	0.725	0.637	0.678	-6.0
Trained with all, except Waco (1263 tiles, 49.0% not persuasive)								
Features	Accuracy	Baseline Accuracy	% Change	Precision	Recall	F-Score	Baseline F-score	% Change
unigrams	0.373	0.7	-46.7	0.285	0.721	0.408	0.462	-11.7
bigrams	0.347	0.7	-50.4	0.296	0.853	0.439	0.462	-5.0
GBGs	0.444	0.7	-36.6	0.326	0.801	0.464	0.462	0.4
OSBs	0.338	0.7	-51.7	0.301	0.912	0.453	0.462	-1.9

Table 4.15: Naive bayes over tiles, trained on three of four transcript types

lowest accuracy and precision, while OSBs resulted in the lowest recall and F-score. In the Taylor set, OSBs produced the highest accuracy and F-score. Bigrams produced the highest recall, while unigrams produced the highest precision.

In other sets of experiments, leaving out the San Diego transcript has dramatically, improved all metrics. However, this not the case in this experiments despite the small size of the transcript (78 tiles, 4.5% of the corpus). Additionally, in this set, gappy bigrams and OSBs resulted in the same accuracies, but different recall and precision. This was also seen in results presented in section 4.9.2. The Waco set generally produced the lowest precision, but the highest recall for each feature set. This set of results provided still more evidence that tiles did not help to successfully detect persuasion. This being the case, no conclusions can be drawn about the different transcript types from these results.

4.9.4 Performance of TextTiling

Since only three experiments outperformed the baseline F-score, TextTiling did not provide a suitable method of segmentation for this classifications task. This set of experiments reiforced

Trained with all, except Rogan (1195 tiles, 55.8% not persuasive)								
Features	Accuracy	Baseline Accuracy	% Change	Precision	Recall	F-Score	Baseline F-score	% Change
unigrams	0.626	0.516	21.3	0.626	0.563	0.593	0.652	-9.0
bigrams	0.566	0.516	9.7	0.57	0.421	0.484	0.652	-25.8
GBGs	0.572	0.516	10.9	0.588	0.385	0.465	0.652	-28.7
OSBs	0.578	0.516	12.0	0.604	0.369	0.458	0.652	-29.8
Trained with all, except Taylor (1052 tiles, 59.3% not persuasive)								
Features	Accuracy	Baseline Accuracy	% Change	Precision	Recall	F-Score	Baseline F-score	% Change
unigrams	0.474	0.53	-10.6	0.714	0.014	0.028	0.693	-96.0
bigrams	0.532	0.53	0.4	0.613	0.315	0.417	0.693	-39.8
GBGs	0.524	0.53	-1.1	0.607	0.29	0.392	0.693	-43.4
OSBs	0.536	0.53	1.1	0.639	0.287	0.396	0.693	-42.9
Trained with all, except SDPolice (1638 tiles, 54.8% not persuasive)								
Features	Accuracy	Baseline Accuracy	% Change	Precision	Recall	F-Score	Baseline F-score	% Change
unigrams	0.551	0.513	7.4	0.561	0.575	0.568	0.678	-16.2
bigrams	0.513	0.513	0.0	0.526	0.5	0.513	0.678	-24.3
GBGs	0.564	0.513	9.9	0.579	0.55	0.564	0.678	-16.8
OSBs	0.564	0.513	9.9	0.583	0.525	0.553	0.678	-18.4
Trained with all, except Waco (1263 tiles, 49.0% not persuasive)								
Features	Accuracy	Baseline Accuracy	% Change	Precision	Recall	F-Score	Baseline F-score	% Change
unigrams	0.558	0.7	-20.3	0.371	0.676	0.479	0.462	3.7
bigrams	0.581	0.7	-17.0	0.376	0.603	0.463	0.462	0.2
GBGs	0.567	0.7	-19.0	0.371	0.632	0.467	0.462	1.1
OSBs	0.57	0.7	-18.6	0.364	0.581	0.448	0.462	-3.0

Table 4.16: Support vector machine over tiles, trained on three of four transcript types

the conclusions in section 4.7.4, by showing that the poor results generalize to this set of experiments.

4.10 Leave-One-In over Posts

The results presented in section 4.8 validated that it is possible to train weak classifiers to recognize persuasion in posts. A review of the results of the experiments conducted using the second method described in section 3.7 reinforced this claim.

4.10.1 Maximum Entropy

The set of results shown in Table 4.17 were produced by experiments using maximum entropy. They illustrate differences between the different transcript types. The experiments trained with only Rogan and with only Taylor produced similar results. Each set of features performed the same with regards to the rank order of the results for each metric. In these experiments, the accuracies were within one a 1% change of the baseline accuracies. Both types of experiments showed stronger precision, than recall.

Trained with only Rogan (5249 posts, 85.0% not persuasive)								
Features	Accuracy	Baseline Accuracy	% Change	Precision	Recall	F-Score	Baseline F-score	% Change
unigrams	0.889	0.897	-0.9	0.458	0.42	0.438	0.187	134.2
bigrams	0.899	0.897	0.2	0.515	0.263	0.348	0.187	86.1
GBGs	0.892	0.897	-0.6	0.465	0.305	0.369	0.187	97.3
OSBs	0.902	0.897	0.6	0.559	0.216	0.312	0.187	66.8
Trained with only Taylor (6913 posts, 87.2% not persuasive)								
Features	Accuracy	Baseline Accuracy	% Change	Precision	Recall	F-Score	Baseline F-score	% Change
unigrams	0.886	0.891	-0.6	0.473	0.366	0.413	0.197	109.6
bigrams	0.896	0.891	0.6	0.551	0.272	0.365	0.197	85.3
GBGs	0.892	0.891	0.1	0.509	0.308	0.384	0.197	94.9
OSBs	0.898	0.891	0.8	0.581	0.242	0.341	0.197	73.1
Trained with only SDPolice (824 posts, 82.9% not persuasive)								
Features	Accuracy	Baseline Accuracy	% Change	Precision	Recall	F-Score	Baseline F-score	% Change
unigrams	0.875	0.887	-1.4	0.438	0.364	0.397	0.204	94.6
bigrams	0.89	0.887	0.3	0.599	0.085	0.148	0.204	-27.5
GBGs	0.888	0.887	0.1	0.524	0.161	0.247	0.204	21.1
OSBs	0.889	0.887	0.2	0.621	0.064	0.116	0.204	-43.1
Trained with only Waco (5871 posts, 93.7% not persuasive)								
Features	Accuracy	Baseline Accuracy	% Change	Precision	Recall	F-Score	Baseline F-score	% Change
unigrams	0.865	0.86	0.6	0.554	0.195	0.288	0.246	17.1
bigrams	0.868	0.86	0.9	0.744	0.087	0.155	0.246	-37.0
GBGs	0.865	0.86	0.6	0.611	0.106	0.181	0.246	-26.4
OSBs	0.865	0.86	0.6	0.766	0.054	0.101	0.246	-58.9

Table 4.17: Maximum entropy over posts, trained on one of four transcript types

The experiments trained with only using the San Diego Police transcript resulted in lower F-scores than the Taylor and Rogan sets across all features. This was due to decreased recall across all feature sets, while precision remained relatively the same or increased. This was an expected result due to the small size of the training set. Training on only Waco produced the lowest F-scores across each feature set, except for bigrams, which was the second lowest F-score. This set of experiments provides still more evidence that the Waco transcripts are different than the others.

4.10.2 Naive Bayes

Table 4.18 contains the results of the naive bayes experiments. The experiments trained without Rogan unigrams resulted in the highest F-score and recall and the second highest accuracy and precision. In the Taylor set, gappy bigrams produced the highest precision, recall, and F-score and the second highest accuracy. While gappy bigrams resulted in higher scores for all metrics, these two sets of experiments were not as similar as previous sets. The experiments trained only using the San Diego Police transcript produce results similar to the Rogan and Taylor experiments. The Waco set generally produced the lowest F-score for each feature set. These

Trained with only Rogan (5249 posts, 85.0% not persuasive)								
Features	Accuracy	Baseline Accuracy	% Change	Precision	Recall	F-Score	Baseline F-score	% Change
unigrams	0.872	0.897	-2.8	0.384	0.399	0.392	0.187	109.6
bigrams	0.845	0.897	-5.8	0.329	0.487	0.393	0.187	110.2
GBGs	0.859	0.897	-4.2	0.373	0.541	0.442	0.187	136.4
OSBs	0.857	0.897	-4.5	0.364	0.521	0.428	0.187	128.9
Trained with only Taylor (6913 posts, 87.2% not persuasive)								
Features	Accuracy	Baseline Accuracy	% Change	Precision	Recall	F-Score	Baseline F-score	% Change
unigrams	0.877	0.891	-1.6	0.426	0.366	0.394	0.197	100.0
bigrams	0.856	0.891	-3.9	0.359	0.402	0.379	0.197	92.4
GBGs	0.878	0.891	-1.5	0.436	0.415	0.426	0.197	116.2
OSBs	0.879	0.891	-1.3	0.433	0.348	0.386	0.197	95.9
Trained with only SDPolice (824 posts, 82.9% not persuasive)								
Features	Accuracy	Baseline Accuracy	% Change	Precision	Recall	F-Score	Baseline F-score	% Change
unigrams	0.843	0.887	-5.0	0.348	0.442	0.39	0.204	91.2
bigrams	0.814	0.887	-8.2	0.312	0.535	0.394	0.204	93.1
GBGs	0.824	0.887	-7.1	0.339	0.578	0.427	0.204	109.3
OSBs	0.803	0.887	-9.5	0.311	0.61	0.412	0.204	102.0
Trained with only Waco (5871 posts, 93.7% not persuasive)								
Features	Accuracy	Baseline Accuracy	% Change	Precision	Recall	F-Score	Baseline F-score	% Change
unigrams	0.856	0.86	-0.5	0.461	0.175	0.254	0.246	3.3
bigrams	0.84	0.86	-2.3	0.292	0.101	0.15	0.246	-39.0
GBGs	0.856	0.86	-0.5	0.368	0.045	0.08	0.246	-67.5
OSBs	0.851	0.86	-1.0	0.335	0.065	0.108	0.246	-56.1

Table 4.18: Naive bayes over posts, trained on one of four transcript types

results contain trend similar to the previous set of results and further reinforce that the Waco transcripts are sufficiently different from the other three types of transcripts.

4.10.3 Support Vector Machine

In the results of the SVM experiments are shown in Table 4.19. The experiments trained with only Rogan and with only Taylor produced similar results. Each set of features performed generally the same with regards to the rank order of the results for accuracy, precision, and recall. In these experiments, the F-scores were higher than the baseline F-scores. Both types of experiments showed stronger precision, than recall.

The experiments trained with only using the San Diego Police transcript resulted in lower F-scores than the Taylor and Rogan sets across all features. Training on only Waco produced the lowest F-scores across each feature set. This is surprising given the size of the training set.

4.10.4 Effects of Leave-One-In on Posts

The results in this section reinforce the claim that the Rogan and Taylor sets are similar since they exhibited similar results when used as training data. Not surprisingly, the San Diego Po-

Trained with only Rogan (5249 posts, 85.0% not persuasive)								
Features	Accuracy	Baseline Accuracy	% Change	Precision	Recall	F-Score	Baseline F-score	% Change
unigrams	0.88	0.897	-1.9	0.416	0.418	0.417	0.187	123.0
bigrams	0.887	0.897	-1.1	0.432	0.323	0.37	0.187	97.9
GBGs	0.895	0.897	-0.2	0.482	0.327	0.39	0.187	108.6
OSBs	0.896	0.897	-0.1	0.49	0.295	0.369	0.187	97.3
Trained with only Taylor (6913 posts, 87.2% not persuasive)								
Features	Accuracy	Baseline Accuracy	% Change	Precision	Recall	F-Score	Baseline F-score	% Change
unigrams	0.881	0.891	-1.1	0.442	0.36	0.397	0.197	101.5
bigrams	0.889	0.891	-0.2	0.49	0.341	0.402	0.197	104.1
GBGs	0.894	0.891	0.3	0.52	0.337	0.409	0.197	107.6
OSBs	0.894	0.891	0.3	0.526	0.306	0.387	0.197	96.4
Trained with only SDPolice (824 posts, 82.9% not persuasive)								
Features	Accuracy	Baseline Accuracy	% Change	Precision	Recall	F-Score	Baseline F-score	% Change
unigrams	0.851	0.887	-4.1	0.358	0.402	0.379	0.204	85.8
bigrams	0.886	0.887	-0.1	0.497	0.216	0.301	0.204	47.5
GBGs	0.888	0.887	0.1	0.518	0.237	0.325	0.204	59.3
OSBs	0.889	0.887	0.2	0.528	0.19	0.279	0.204	36.8
Trained with only Waco (5871 posts, 93.7% not persuasive)								
Features	Accuracy	Baseline Accuracy	% Change	Precision	Recall	F-Score	Baseline F-score	% Change
unigrams	0.86	0.86	0.0	0.499	0.234	0.318	0.246	29.3
bigrams	0.866	0.86	0.7	0.616	0.108	0.184	0.246	-25.2
GBGs	0.867	0.86	0.8	0.669	0.104	0.18	0.246	-26.8
OSBs	0.868	0.86	0.9	0.741	0.087	0.155	0.246	-37.0

Table 4.19: Support vector machine over posts, trained on one of four transcript types

lice experiments produced worse results than the Rogan and Taylor experiments for maximum entropy and support vector machine. This change in performance can mainly be attributed to the small size of the transcript(824 posts, 4.4% of the corpus).

However, for naive bayes, the San Diego experiments performed comparably with the Rogan and Taylor experiments. This can also be attributed to the small training set. Ng and Jordan proved that generative models, like naive bayes, reach their asymptotic error more quickly, than discriminative models [34]. Discriminative models, such as maximum entropy and support vector machine, outperform generative models for larger data sets. The fact that recall and precision are the same for a small data set suggests that there are certain lexical features that are strong indicators of persuasions.

Again, the Waco experiments produced significantly different results than the other experiments. This further reinforces the hypothesis that the Waco transcripts are significantly different from the other three types.

4.11 Leave-One-In over Tiles

The results presented in section 4.9 exhibited the same trends as in section 4.7. The results of the experiments conducted using the second method described in section 3.7 continued to strengthen the claim that no signal is learned when using tiles.

4.11.1 Maximum Entropy

Trained with only Rogan (521 tiles, 51.6% not persuasive)								
Features	Accuracy	Baseline Accuracy	% Change	Precision	Recall	F-Score	Baseline F-score	% Change
unigrams	0.527	0.558	-5.6	0.46	0.398	0.426	0.613	-30.5
bigrams	0.502	0.558	-10.0	0.452	0.598	0.515	0.613	-16.0
GBGs	0.515	0.558	-7.7	0.461	0.576	0.512	0.613	-16.5
OSBs	0.488	0.558	-12.5	0.45	0.708	0.55	0.613	-10.3
Trained with only Taylor (664 tiles, 47.0% not persuasive)								
Features	Accuracy	Baseline Accuracy	% Change	Precision	Recall	F-Score	Baseline F-score	% Change
unigrams	0.573	0.593	-3.4	0.472	0.409	0.438	0.578	-24.2
bigrams	0.501	0.593	-15.5	0.429	0.685	0.527	0.578	-8.8
GBGs	0.482	0.593	-18.7	0.427	0.794	0.555	0.578	-4.0
OSBs	0.436	0.593	-26.5	0.413	0.921	0.571	0.578	-1.2
Trained with only SDPolice (78 tiles, 48.7% not persuasive)								
Features	Accuracy	Baseline Accuracy	% Change	Precision	Recall	F-Score	Baseline F-score	% Change
unigrams	0.53	0.548	-3.3	0.482	0.546	0.512	0.622	-17.7
bigrams	0.493	0.548	-10.0	0.467	0.857	0.604	0.622	-2.9
GBGs	0.501	0.548	-8.6	0.472	0.884	0.615	0.622	-1.1
OSBs	0.475	0.548	-13.3	0.46	0.942	0.618	0.622	-0.6
Trained with only Waco (453 tiles, 70% not persuasive)								
Features	Accuracy	Baseline Accuracy	% Change	Precision	Recall	F-Score	Baseline F-score	% Change
unigrams	0.515	0.51	1.0	0.534	0.379	0.443	0.675	-34.4
bigrams	0.49	0.51	-3.9	0.5	0.005	0.009	0.675	-98.7
GBGs	0.489	0.51	-4.1	0.4	0.003	0.006	0.675	-99.1
OSBs	0.49	0.51	-3.9	0.0	0.0	0.0	0.675	-100.0

Table 4.20: Maximum entropy over tiles, trained on one of four transcript types

Table 4.20 shows the results of the maximum entropy experiments. OSBs produced the best F-scores for the Rogan, Taylor and San Diego Police experiments, while unigrams produced the lowest F-score. The performance of the other two types of bigrams were similar to OSBs. The disparity in performance among the features can be explained by the parameter values for λ. The unigram experiments had a small λ value (2^{-7}), while the three types of bigram experiments had larger λ values (2^{8} or 2^{10}). The Waco experiments classified nearly all tiles as not persuasive, with the exception of the unigram experiment. Again, this can be explained by the difference in λ values. The poor performance of all experiments continues to reinforce the ineffectiveness of tiles for this classification task.

Trained with only Rogan (521 tiles, 51.6% not persuasive)								
Features	Accuracy	Baseline Accuracy	% Change	Precision	Recall	F-Score	Baseline F-score	% Change
unigrams	0.481	0.558	-13.8	0.426	0.5	0.46	0.613	-25.0
bigrams	0.505	0.558	-9.5	0.463	0.756	0.575	0.613	-6.2
GBGs	0.529	0.558	-5.2	0.476	0.646	0.548	0.613	-10.6
OSBs	0.513	0.558	-8.1	0.467	0.733	0.571	0.613	-6.9
Trained with only Taylor (664 tiles, 47.0% not persuasive)								
Features	Accuracy	Baseline Accuracy	% Change	Precision	Recall	F-Score	Baseline F-score	% Change
unigrams	0.434	0.593	-26.8	0.408	0.862	0.554	0.578	-4.2
bigrams	0.446	0.593	-24.8	0.416	0.895	0.568	0.578	-1.7
GBGs	0.457	0.593	-22.9	0.417	0.841	0.558	0.578	-3.5
OSBs	0.419	0.593	-29.3	0.409	0.958	0.573	0.578	-0.9
Trained with only SDPolice (78 tiles, 48.7% not persuasive)								
Features	Accuracy	Baseline Accuracy	% Change	Precision	Recall	F-Score	Baseline F-score	% Change
unigrams	0.462	0.548	-15.7	0.452	0.905	0.603	0.622	-3.1
bigrams	0.464	0.548	-15.3	0.453	0.904	0.604	0.622	-2.9
GBGs	0.488	0.548	-10.9	0.465	0.892	0.611	0.622	-1.8
OSBs	0.471	0.548	-14.1	0.458	0.926	0.612	0.622	-1.6
Trained with only Waco (453 tiles, 70% not persuasive)								
Features	Accuracy	Baseline Accuracy	% Change	Precision	Recall	F-Score	Baseline F-score	% Change
unigrams	0.465	0.51	-8.8	0.468	0.363	0.409	0.675	-39.4
bigrams	0.479	0.51	-6.1	0.469	0.166	0.245	0.675	-63.7
GBGs	0.496	0.51	-2.7	0.585	0.037	0.07	0.675	-89.6
OSBs	0.488	0.51	-4.3	0.452	0.022	0.041	0.675	-93.9

Table 4.21: Naive bayes over tiles, trained on one of four transcript types

4.11.2 Naive Bayes

The results of the naive bayes experiments are shown in Table 4.21, bigrams produced the best F-scores for the Rogan experiments. The Taylor and San Diego Police experiments showed their highest F-score when using OSBs. The Waco experiments exhibited a maximum F-score when using unigrams. The Taylor and San Diego experiments often called all tiles persuasive, resulting in higher recall with precision very similar to accuracy. The Rogan experiments showed the same trend, but less extreme. The Waco experiments showed the opposite trend, predicting not persuasive for most tiles. Yet, again, tiles continued to be ineffective.

4.11.3 Support Vector Machine

Table 4.22 shows the results of the SVM experiments. Unigrams produced the best F-scores for the Rogan, Taylor and San Diego Police experiments. Bigrams produced the lowest F-scores for the Taylor and San Diego experiments, and the second lowest F-score for the Rogan experiments. The Waco experiments classified all tiles as not persuasive, with the exception of the bigram experiment where a few classified as persuasive.

Trained with only Rogan (521 tiles, 51.6% not persuasive)								
Features	Accuracy	Baseline Accuracy	% Change	Precision	Recall	F-Score	Baseline F-score	% Change
unigrams	0.573	0.558	2.7	0.515	0.589	0.549	0.613	-10.4
bigrams	0.572	0.558	2.5	0.515	0.519	0.517	0.613	-15.7
GBGs	0.573	0.558	2.7	0.516	0.547	0.531	0.613	-13.4
OSBs	0.57	0.558	2.2	0.513	0.511	0.512	0.613	-16.5
Trained with only Taylor (664 tiles, 47.0% not persuasive)								
Features	Accuracy	Baseline Accuracy	% Change	Precision	Recall	F-Score	Baseline F-score	% Change
unigrams	0.548	0.593	-7.6	0.466	0.764	0.579	0.578	0.2
bigrams	0.571	0.593	-3.7	0.48	0.638	0.548	0.578	-5.2
GBGs	0.544	0.593	-8.3	0.46	0.694	0.553	0.578	-4.3
OSBs	0.564	0.593	-4.9	0.475	0.692	0.563	0.578	-2.6
Trained with only SDPolice (78 tiles, 48.7% not persuasive)								
Features	Accuracy	Baseline Accuracy	% Change	Precision	Recall	F-Score	Baseline F-score	% Change
unigrams	0.452	0.548	-17.5	0.452	1.0	0.622	0.622	0.0
bigrams	0.57	0.548	4.0	0.523	0.549	0.535	0.622	-14.0
GBGs	0.559	0.548	2.0	0.51	0.63	0.563	0.622	-9.5
OSBs	0.56	0.548	2.2	0.511	0.641	0.568	0.622	-8.7
Trained with only Waco (453 tiles, 70% not persuasive)								
Features	Accuracy	Baseline Accuracy	% Change	Precision	Recall	F-Score	Baseline F-score	% Change
unigrams	0.49	0.51	-3.9	0.0	0.0	0.0	0.675	-100.0
bigrams	0.492	0.51	-3.5	0.75	0.005	0.009	0.675	-98.7
GBGs	0.49	0.51	-3.9	0.0	0.0	0.0	0.675	-100.0
OSBs	0.49	0.51	-3.9	0.0	0.0	0.0	0.675	-100.0

Table 4.22: Support vector machine over tiles, trained on one of four transcript types

4.11.4 Performance of TextTiling

The results in this section continued to verify the poor performance of tiles for this class. There are no new conclusions that can be drawn from this set of experiments.

4.12 Leave-One-Out over Posts without Waco

All results from experiments using posts indicated that the Waco transcripts are sufficiently different from the other three types of transcripts. In order to investigate this claim, the experiments described in section 3.7 were repeated without including the Waco transcripts. The results of the leave-one-out experiments are presented in this section.

4.12.1 Maximum Entropy

The results of maximum entropy experiments are contained in Table 4.23. The experiments trained without Rogan and without Taylor produced similar results. Each set of features performed the same with regards to the rank order of the results for each metric. Both types of experiments showed stronger precision, than recall. The experiments trained without using the San Diego Police transcript resulted in the highest scores for each metric.

Trained with all, except Rogan (7737 posts, 86.7% not persuasive)								
Features	Accuracy	Baseline Accuracy	% Change	Precision	Recall	F-Score	Baseline F-score	% Change
unigrams	0.865	0.85	1.8	0.58	0.379	0.458	0.262	74.8
bigrams	0.869	0.85	2.2	0.656	0.272	0.385	0.262	46.9
GBGs	0.868	0.85	2.1	0.617	0.328	0.428	0.262	63.4
OSBs	0.874	0.85	2.8	0.721	0.263	0.385	0.262	46.9
Trained with all, except Taylor (6073 posts, 84.7% not persuasive)								
Features	Accuracy	Baseline Accuracy	% Change	Precision	Recall	F-Score	Baseline F-score	% Change
unigrams	0.875	0.872	0.3	0.516	0.411	0.458	0.227	101.8
bigrams	0.882	0.872	1.1	0.587	0.251	0.352	0.227	55.1
GBGs	0.878	0.872	0.7	0.543	0.298	0.385	0.227	69.6
OSBs	0.881	0.872	1.0	0.599	0.205	0.305	0.227	34.4
Trained with all, except SDPolice (12162 posts, 86.3% not persuasive)								
Features	Accuracy	Baseline Accuracy	% Change	Precision	Recall	F-Score	Baseline F-score	% Change
unigrams	0.897	0.82	9.4	0.774	0.582	0.664	0.297	123.6
bigrams	0.905	0.82	10.4	0.848	0.553	0.67	0.297	125.6
GBGs	0.895	0.82	9.1	0.804	0.525	0.635	0.297	113.8
OSBs	0.897	0.82	9.4	0.872	0.482	0.621	0.297	109.1

Table 4.23: Maximum entropy over posts, trained on two of three transcript types **(Waco not included)**

4.12.2 Naive Bayes

Trained with all, except Rogan (7737 posts, 86.7% not persuasive)								
Features	Accuracy	Baseline Accuracy	% Change	Precision	Recall	F-Score	Baseline F-score	% Change
unigrams	0.857	0.85	0.8	0.538	0.352	0.425	0.262	62.2
bigrams	0.839	0.85	-1.3	0.464	0.449	0.456	0.262	74.0
GBGs	0.859	0.85	1.1	0.533	0.488	0.509	0.262	94.3
OSBs	0.859	0.85	1.1	0.539	0.426	0.476	0.262	81.7
Trained with all, except Taylor (6073 posts, 84.7% not persuasive)								
Features	Accuracy	Baseline Accuracy	% Change	Precision	Recall	F-Score	Baseline F-score	% Change
unigrams	0.862	0.872	-1.1	0.453	0.374	0.41	0.227	80.6
bigrams	0.841	0.872	-3.6	0.403	0.508	0.45	0.227	98.2
GBGs	0.851	0.872	-2.4	0.436	0.562	0.491	0.227	116.3
OSBs	0.852	0.872	-2.3	0.437	0.547	0.486	0.227	114.1
Trained with all, except SDPolice (12162 posts, 86.3% not persuasive)								
Features	Accuracy	Baseline Accuracy	% Change	Precision	Recall	F-Score	Baseline F-score	% Change
unigrams	0.902	0.826	9.2	0.777	0.617	0.688	0.297	131.6
bigrams	0.889	0.826	7.6	0.711	0.61	0.656	0.297	120.9
GBGs	0.91	0.826	10.2	0.743	0.738	0.74	0.297	149.2
OSBs	0.904	0.826	9.4	0.769	0.638	0.698	0.297	135.0

Table 4.24: Naive bayes over posts, trained on two of three transcript types **(Waco not included)**

Table 4.24 shows the results of the naive bayes experiments. The experiments trained without Rogan and without Taylor produced similar results. Each set of features performed the same with regards to the rank order of F-scores. The Rogan experiments showed stronger precision, than recall. The reverse was true for the Taylor experiments. The experiments trained without using the San Diego Police transcript boasted the highest scores for each metric.

4.12.3 Support Vector Machine

Trained with all, except Rogan (7737 posts, 86.7% not persuasive)								
Features	Accuracy	Baseline Accuracy	% Change	Precision	Recall	F-Score	Baseline F-score	% Change
unigrams	0.843	0.85	-0.8	0.471	0.351	0.402	0.262	53.4
bigrams	0.865	0.85	1.8	0.586	0.351	0.439	0.262	67.6
GBGs	0.871	0.85	2.5	0.631	0.343	0.444	0.262	69.5
OSBs	0.871	0.85	2.5	0.641	0.324	0.43	0.262	64.1
Trained with all, except Taylor (6073 posts, 84.7% not persuasive)								
Features	Accuracy	Baseline Accuracy	% Change	Precision	Recall	F-Score	Baseline F-score	% Change
unigrams	0.866	0.872	-0.7	0.473	0.407	0.437	0.227	92.5
bigrams	0.876	0.872	0.5	0.526	0.337	0.41	0.227	80.6
GBGs	0.882	0.872	1.1	0.569	0.33	0.418	0.227	84.1
OSBs	0.882	0.872	1.1	0.573	0.292	0.387	0.227	70.5
Trained with all, except SDPolice (12162 posts, 86.3% not persuasive)								
Features	Accuracy	Baseline Accuracy	% Change	Precision	Recall	F-Score	Baseline F-score	% Change
unigrams	0.894	0.826	8.2	0.767	0.56	0.648	0.297	118.2
bigrams	0.906	0.826	9.7	0.798	0.617	0.696	0.297	134.3
GBGs	0.912	0.826	10.4	0.85	0.603	0.705	0.297	137.4
OSBs	0.896	0.826	8.5	0.806	0.532	0.641	0.297	115.8

Table 4.25: Support vector machine over posts, trained on two of three transcript types **(Waco not included)**

In the set of experiments shown in Table 4.25, the trend that experiments training by leaving out San Diego Police, performed the best, followed by leaving out Rogan, continued. The experiments trained without Rogan and without Taylor produced similar results. Each set of features performed the same with regards to the rank order of the results for accuracy, precision, and recall.

4.12.4 Effects of Eliminating Waco for Posts

In general, each set of experiments produced a decrease in precision with an increase in recall. However, these changes were not as significant as we anticipated. While most F-scores increased by more than .01, several had less than a .01 change and one F-score decreased by more than .01. The next section presents of these same experiments, conducted using tiles.

4.13 Leave-One-Out over Tiles without Waco

All results from experiments using tiles indicated that no signal for persuasion was learned. In order to full investigate this claim, it is necessary to verify that this is not caused by the Waco transcript set. This was accomplished by repeating the experiments described in section 3.7. The results of these experiments are presented in this section.

Trained with all, except Rogan (742 tiles, 47.2% not persuasive)								
Features	Accuracy	Baseline Accuracy	% Change	Precision	Recall	F-Score	Baseline F-score	% Change
unigrams	0.56	0.516	8.5	0.55	0.5	0.524	0.652	-19.6
bigrams	0.518	0.516	0.4	0.501	0.726	0.593	0.652	-9.0
GBGs	0.518	0.516	0.4	0.501	0.817	0.621	0.652	-4.8
OSBs	0.509	0.516	-1.4	0.496	0.921	0.644	0.652	-1.2
Trained with all, except Taylor (599 tiles, 51.3% not persuasive)								
Features	Accuracy	Baseline Accuracy	% Change	Precision	Recall	F-Score	Baseline F-score	% Change
unigrams	0.523	0.507	3.2	0.56	0.463	0.507	0.693	-26.8
bigrams	0.545	0.507	7.5	0.564	0.622	0.592	0.693	-14.6
GBGs	0.547	0.507	7.9	0.561	0.67	0.611	0.693	-11.8
OSBs	0.523	0.507	3.2	0.534	0.778	0.634	0.693	-8.5
Trained with all, except SDPolice (1185 tiles, 49.0% not persuasive)								
Features	Accuracy	Baseline Accuracy	% Change	Precision	Recall	F-Score	Baseline F-score	% Change
unigrams	0.59	0.513	15.0	0.6	0.6	0.6	0.678	-11.5
bigrams	0.641	0.513	25.0	0.62	0.775	0.689	0.678	1.6
GBGs	0.59	0.513	15.0	0.58	0.725	0.644	0.678	-5.0
OSBs	0.603	0.513	17.5	0.574	0.875	0.693	0.678	2.2

Table 4.26: Maximum entropy over tiles, trained on two of three transcript types **(Waco not included)**

4.13.1 Maximum Entropy

Table 4.26 contains the results of the maximum entropy experiments. The experiments trained without Rogan and without Taylor produced similar results when using OSBs as features, but not other feature sets. The experiments trained without using the San Diego Police transcript resulted in the highest scores for each metric within each type of feature.

4.13.2 Naive Bayes

The results of the naive bayes experiments are shown in Table 4.27 The experiments trained without Rogan and without Taylor produced similar results. Each set of features performed the same with regards to the rank order of the results for each metric. Both types of experiments showed stronger recall, than precision. The experiments trained without using the San Diego Police transcript resulted in the highest scores for each metric.

4.13.3 Support Vector Machine

The results of the SVM experiments are contained in Table 4.28. The experiments trained without Rogan and without Taylor produced similar results. However, the San Diego Police experiments did not produce significantly better results despite the fact that most of the data was used for training.

Trained with all, except Rogan (742 tiles, 47.2% not persuasive								
Features	Accuracy	Baseline Accuracy	% Change	Precision	Recall	F-Score	Baseline F-score	% Change
unigrams	0.47	0.516	-8.9	0.474	0.853	0.609	0.652	-6.6
bigrams	0.493	0.516	-4.5	0.487	0.921	0.637	0.652	-2.3
GBGs	0.501	0.516	-2.9	0.491	0.889	0.633	0.652	-2.9
OSBs	0.497	0.516	-3.7	0.49	0.968	0.651	0.652	-0.2
Trained with all, except Taylor (599 tiles, 51.3% not persuasive)								
Features	Accuracy	Baseline Accuracy	% Change	Precision	Recall	F-Score	Baseline F-score	% Change
unigrams	0.486	0.53	-8.3	0.515	0.545	0.53	0.693	-23.5
bigrams	0.511	0.53	-3.6	0.525	0.807	0.636	0.693	-8.2
GBGs	0.541	0.53	2.1	0.548	0.767	0.639	0.693	-7.8
OSBs	0.53	0.53	0.0	0.536	0.838	0.654	0.693	-5.6
Trained with all, except SDPolice (1185 tiles, 49.0% not persuasive)								
Features	Accuracy	Baseline Accuracy	% Change	Precision	Recall	F-Score	Baseline F-score	% Change
unigrams	0.551	0.513	7.4	0.537	0.9	0.673	0.678	-0.7
bigrams	0.538	0.513	4.9	0.529	0.925	0.673	0.678	-0.7
GBGs	0.603	0.513	17.5	0.569	0.925	0.705	0.678	4.0
OSBs	0.474	0.513	-7.6	0.493	0.9	0.637	0.678	-6.0

Table 4.27: Naive bayes over tiles, trained on two of three transcript types **(Waco not included)**

Trained with all, except Rogan (742 tiles, 47.2% not persuasive)								
Features	Accuracy	Baseline Accuracy	% Change	Precision	Recall	F-Score	Baseline F-score	% Change
unigrams	0.572	0.516	10.9	0.541	0.762	0.633	0.652	-2.9
bigrams	0.562	0.516	8.9	0.542	0.611	0.575	0.652	-11.8
GBGs	0.547	0.516	6.0	0.524	0.69	0.596	0.652	-8.6
OSBs	0.562	0.516	8.9	0.538	0.679	0.6	0.652	-8.0
Trained with all, except Taylor (599 tiles, 51.3% not persuasive)								
Features	Accuracy	Baseline Accuracy	% Change	Precision	Recall	F-Score	Baseline F-score	% Change
unigrams	0.569	0.53	7.4	0.589	0.622	0.605	0.693	-12.7
bigrams	0.562	0.53	6.0	0.603	0.506	0.55	0.693	-20.6
GBGs	0.557	0.53	5.1	0.588	0.551	0.569	0.693	-17.9
OSBs	0.553	0.53	4.3	0.59	0.514	0.549	0.693	-20.8
Trained with all, except SDPolice (1185 tiles, 49.0% not persuasive)								
Features	Accuracy	Baseline Accuracy	% Change	Precision	Recall	F-Score	Baseline F-score	% Change
unigrams	0.538	0.513	4.9	0.54	0.675	0.6	0.678	-11.5
bigrams	0.564	0.513	9.9	0.568	0.625	0.595	0.678	-12.2
GBGs	0.564	0.513	9.9	0.565	0.65	0.605	0.678	-10.8
OSBs	0.5	0.513	-2.5	0.511	0.6	0.552	0.678	-18.6

Table 4.28: Support vector machine over tiles, trained on two of three transcript types **(Waco not included)**

4.13.4 Effects of Eliminating Waco for Tiles

In general, each set of experiments produced a decrease in precision with an increase in recall. However, these changes were not as significant as we anticipated. While most F-scores increased by more than .01, several had less than a .01 change and three F-scores decreased by more than .01. Additionally, the changes in the results for the two discriminative methods where more significant than the changes for naive bayes. However, since only two experiments

outperformed the baseline F-score, the significance of these observations is unclear. The only new conclusion that can be drawn from this set of experiments is that the Waco transcripts were not the source of the poor performance of tiles.

4.14 Leave-One-In over Posts without Waco

Section 4.12 contains results that reinforced the claim that the Waco transcripts are sufficiently different from the other three types of transcripts. It was possible to verify this claim further by repeating the leave-one-in experiments described section 3.7 without the Waco transcripts. The results of these experiments are presented in this section.

4.14.1 Maximum Entropy

Trained with only Rogan (5249 posts, 85.0% not persuasive)								
Features	Accuracy	Baseline Accuracy	% Change	Precision	Recall	F-Score	Baseline F-score	% Change
unigrams	0.875	0.867	0.9	0.536	0.444	0.486	0.235	106.8
bigrams	0.879	0.867	1.4	0.604	0.269	0.372	0.235	58.3
GBGs	0.876	0.867	1.0	0.56	0.309	0.398	0.235	69.4
OSBs	0.881	0.867	1.6	0.658	0.223	0.333	0.235	41.7
Trained with only Taylor (6913 posts, 87.2% not persuasive)								
Features	Accuracy	Baseline Accuracy	% Change	Precision	Recall	F-Score	Baseline F-score	% Change
unigrams	0.866	0.846	2.4	0.595	0.401	0.479	0.266	80.1
bigrams	0.873	0.846	3.2	0.712	0.286	0.408	0.266	53.4
GBGs	0.873	0.846	3.2	0.674	0.332	0.445	0.266	67.3
OSBs	0.874	0.846	3.3	0.764	0.259	0.386	0.266	45.1
Trained with only SDPolice (824 posts, 82.9% not persuasive)								
Features	Accuracy	Baseline Accuracy	% Change	Precision	Recall	F-Score	Baseline F-score	% Change
unigrams	0.862	0.862	0.0	0.501	0.377	0.43	0.242	77.7
bigrams	0.869	0.862	0.8	0.756	0.074	0.135	0.242	-44.2
GBGs	0.87	0.862	0.9	0.613	0.161	0.255	0.242	5.4
OSBs	0.868	0.862	0.7	0.754	0.059	0.109	0.242	-55.0

Table 4.29: Maximum entropy over posts, trained on one of three transcript types **(Waco not included)**

Table 4.29 shows the results of the maximum entropy experiments. The experiments trained with only Rogan and only Taylor produced similar results. Each set of features performed the same with regards to the rank order of the results for each metric. Both types of experiments showed stronger precision, than recall. The experiments trained using only the San Diego Police transcript resulted in the lowest scores accuracy, recall, and F-score. Some precision scores were higher, but these scores were accompanied by low recall.

4.14.2 Naive Bayes

Table 4.30 contains the results of the naive bayes experiments. The experiments trained with only Rogan produced their highest F-score using gappy bigrams and their lowest F-score using

Trained with only Rogan (5249 posts, 85.0% not persuasive)								
Features	Accuracy	Baseline Accuracy	% Change	Precision	Recall	F-Score	Baseline F-score	% Change
unigrams	0.864	0.867	-0.3	0.483	0.378	0.424	0.235	80.4
bigrams	0.839	0.867	-3.2	0.408	0.473	0.438	0.235	86.4
GBGs	0.86	0.867	-0.8	0.476	0.538	0.505	0.235	114.9
OSBs	0.856	0.867	-1.3	0.463	0.519	0.489	0.235	108.1
Trained with only Taylor (6913 posts, 87.2% not persuasive)								
Features	Accuracy	Baseline Accuracy	% Change	Precision	Recall	F-Score	Baseline F-score	% Change
unigrams	0.86	0.846	1.7	0.574	0.356	0.439	0.266	65.0
bigrams	0.841	0.846	-0.6	0.48	0.408	0.441	0.266	65.8
GBGs	0.865	0.846	2.2	0.58	0.429	0.493	0.266	85.3
OSBs	0.859	0.846	1.5	0.565	0.357	0.437	0.266	64.3
Trained with only SDPolice (824 posts, 82.9% not persuasive)								
Features	Accuracy	Baseline Accuracy	% Change	Precision	Recall	F-Score	Baseline F-score	% Change
unigrams	0.832	0.862	-3.5	0.399	0.442	0.42	0.242	73.6
bigrams	0.803	0.862	-6.8	0.357	0.541	0.43	0.242	77.7
GBGs	0.819	0.862	-5.0	0.395	0.589	0.473	0.242	95.5
OSBs	0.794	0.862	-7.9	0.357	0.622	0.454	0.242	87.6

Table 4.30: Naive bayes over posts, trained on one of three transcript types **(Waco not included)**

unigrams. The Taylor experiments produced their highest F-score using gappy bigrams and the lowest F-score using OSBs, followed closely by unigrams. The Taylor experiments showed stronger precision, than recall (excepting unigrams). The reverse was true for the Rogan experiments with the exception of unigrams. The experiments trained without using the San Diego Police transcript generally resulted in the lowest scores for each metric across the feature sets.

4.14.3 Support Vector Machine

The results of the SVM experiments are shown in Table 4.31. The trend that experiments training by leaving out San Diego Police performed the worst continued. The experiments trained with only Rogan and with only Taylor produced similar results. Each set of features performed the same with regards to the rank order of the results for accuracy, precision, and recall.

4.14.4 Effects of Eliminating Waco and Single Transcript Type Training for Posts

In general, each set of experiments produced increases in precision, recall and F-score, when compared to the results in section 4.8. The improved scores were more significant for the Taylor experiments than for the Rogan experiments. The result from this section provided still further evidence than Waco transcripts are substantially different than the other three types of transcripts. The better performance of the Taylor experiments leads us to believe that the data

Trained with only Rogan (5249 posts, 85.0% not persuasive)								
Features	Accuracy	Baseline Accuracy	% Change	Precision	Recall	F-Score	Baseline F-score	% Change
unigrams	0.867	0.867	0.0	0.5	0.438	0.467	0.235	98.7
bigrams	0.872	0.867	0.6	0.529	0.348	0.42	0.235	78.7
GBGs	0.883	0.867	1.8	0.605	0.345	0.439	0.235	86.8
OSBs	0.882	0.867	1.7	0.609	0.32	0.419	0.235	78.3
Trained with only Taylor (6913 posts, 87.2% not persuasive)								
Features	Accuracy	Baseline Accuracy	% Change	Precision	Recall	F-Score	Baseline F-score	% Change
unigrams	0.856	0.846	1.2	0.543	0.39	0.454	0.266	70.7
bigrams	0.871	0.846	3.0	0.637	0.369	0.467	0.266	75.6
GBGs	0.875	0.846	3.4	0.673	0.363	0.472	0.266	77.4
OSBs	0.875	0.846	3.4	0.692	0.337	0.454	0.266	70.7
Trained with only SDPolice (824 posts, 82.9% not persuasive)								
Features	Accuracy	Baseline Accuracy	% Change	Precision	Recall	F-Score	Baseline F-score	% Change
unigrams	0.839	0.862	-2.7	0.414	0.412	0.413	0.242	70.7
bigrams	0.872	0.862	1.2	0.593	0.221	0.322	0.242	33.1
GBGs	0.874	0.862	1.4	0.608	0.237	0.342	0.242	41.3
OSBs	0.874	0.862	1.4	0.633	0.196	0.3	0.242	24.0

Table 4.31: Support vector machine over posts, trained on one of three transcript types **(Waco not included)**

in the Taylor transcripts captures a more easily generalized model. The next section presents of these same experiments, conducted using tiles.

4.15 Leave-One-In over Tiles without Waco

Section 4.13 contains results that continued to validate the hypothesis that no signal was learned from tiles. In addition, the results presented in section 4.13 eliminate the Waco transcript as a possible explanation. For the sake of completeness, the leave-one-in experiments described in section 3.7 were repeated without the Waco transcripts. The results of these experiments are presented in this section.

4.15.1 Maximum Entropy

Table 4.32 shows the results of the maximum entropy experiments. The experiments trained with only Rogan and only Taylor produced their highest F-score using OSBs and their lowest F-score using unigrams. Both sets had stronger recall than precision, with the exception of unigrams. The experiments trained without using the San Diego Police transcript generally resulted in the highest F-scores, but this is misleading. These high F-scores were achieved by calling almost all of the test set persuasive.

Trained with only Rogan (521 tiles, 51.6% not persuasive)								
Features	Accuracy	Baseline Accuracy	% Change	Precision	Recall	F-Score	Baseline F-score	% Change
unigrams	0.503	0.528	-4.7	0.542	0.38	0.447	0.691	-35.3
bigrams	0.524	0.528	-0.8	0.545	0.599	0.571	0.691	-17.4
GBGs	0.491	0.528	-7.0	0.515	0.615	0.56	0.691	-19.0
OSBs	0.507	0.528	-4.0	0.523	0.74	0.613	0.691	-11.3
Trained with only Taylor (664 tiles, 47.0% not persuasive)								
Features	Accuracy	Baseline Accuracy	% Change	Precision	Recall	F-Score	Baseline F-score	% Change
unigrams	0.543	0.513	5.8	0.538	0.432	0.479	0.655	-26.9
bigrams	0.539	0.513	5.1	0.52	0.716	0.602	0.655	-8.1
GBGs	0.529	0.513	3.1	0.511	0.812	0.627	0.655	-4.3
OSBs	0.509	0.513	-0.8	0.498	0.925	0.647	0.655	-1.2
Trained with only SDPolice (78 tiles, 48.7% not persuasive)								
Features	Accuracy	Baseline Accuracy	% Change	Precision	Recall	F-Score	Baseline F-score	% Change
unigrams	0.543	0.51	6.5	0.552	0.541	0.547	0.675	-19.0
bigrams	0.526	0.51	3.1	0.521	0.866	0.65	0.675	-3.7
GBGs	0.54	0.51	5.9	0.529	0.884	0.662	0.675	-1.9
OSBs	0.514	0.51	0.8	0.513	0.942	0.664	0.675	-1.6

Table 4.32: Maximum entropy over tiles, trained on one of three transcript types **(Waco not included)**

Trained with only Rogan (521 tiles, 51.6% not persuasive)								
Features	Accuracy	Baseline Accuracy	% Change	Precision	Recall	F-Score	Baseline F-score	% Change
unigrams	0.485	0.528	-8.1	0.512	0.531	0.521	0.691	-24.6
bigrams	0.522	0.528	-1.1	0.532	0.778	0.632	0.691	-8.5
GBGs	0.522	0.528	-1.1	0.539	0.653	0.591	0.691	-14.5
OSBs	0.527	0.528	-0.2	0.536	0.786	0.637	0.691	-7.8
Trained with only Taylor (664 tiles, 47.0% not persuasive)								
Features	Accuracy	Baseline Accuracy	% Change	Precision	Recall	F-Score	Baseline F-score	% Change
unigrams	0.497	0.513	-3.1	0.491	0.846	0.621	0.655	-5.2
bigrams	0.496	0.513	-3.3	0.491	0.897	0.634	0.655	-3.2
GBGs	0.514	0.513	0.2	0.501	0.836	0.626	0.655	-4.4
OSBs	0.499	0.513	-2.7	0.493	0.966	0.653	0.655	-0.3
Trained with only SDPolice (78 tiles, 48.7% not persuasive)								
Features	Accuracy	Baseline Accuracy	% Change	Precision	Recall	F-Score	Baseline F-score	% Change
unigrams	0.513	0.51	0.6	0.513	0.907	0.655	0.675	-3.0
bigrams	0.505	0.51	-1.0	0.508	0.902	0.65	0.675	-3.7
GBGs	0.527	0.51	3.3	0.521	0.876	0.653	0.675	-3.3
OSBs	0.512	0.51	0.4	0.512	0.927	0.66	0.675	-2.2

Table 4.33: Naive bayes over tiles, trained on one of three transcript types **(Waco not included)**

4.15.2 Naive Bayes

The results of the naive bayes experiments are shown in Table 4.33. The experiments trained with only Rogan and with only Taylor produced similar results with regard to F-score. Each set of features resulted in the same the rank order of F-scores. Both types of experiments showed stronger recall, than precision. The experiments trained with only using the San Diego Police transcript resulted in the highest scores for each metric. The experiments trained without using

the San Diego Police transcript generally resulted in the highest F-scores, but this is misleading. These high F-scores were achieved by calling almost all of the test set persuasive.

4.15.3 Support Vector Machine

Trained with only Rogan (521 tiles, 51.6% not persuasive)								
Features	Accuracy	Baseline Accuracy	% Change	Precision	Recall	F-Score	Baseline F-score	% Change
unigrams	0.553	0.528	4.7	0.575	0.587	0.581	0.691	-15.9
bigrams	0.542	0.528	2.7	0.578	0.492	0.532	0.691	-23.0
GBGs	0.547	0.528	3.6	0.577	0.538	0.557	0.691	-19.4
OSBs	0.536	0.528	1.5	0.57	0.497	0.531	0.691	-23.2
Trained with only Taylor (664 tiles, 47.0% not persuasive)								
Features	Accuracy	Baseline Accuracy	% Change	Precision	Recall	F-Score	Baseline F-score	% Change
unigrams	0.574	0.513	11.9	0.546	0.75	0.632	0.655	-3.5
bigrams	0.578	0.513	12.7	0.558	0.64	0.596	0.655	-9.0
GBGs	0.563	0.513	9.7	0.54	0.688	0.605	0.655	-7.6
OSBs	0.573	0.513	11.7	0.548	0.702	0.616	0.655	-6.0
Trained with only SDPolice (78 tiles, 48.7% not persuasive)								
Features	Accuracy	Baseline Accuracy	% Change	Precision	Recall	F-Score	Baseline F-score	% Change
unigrams	0.51	0.51	0.0	0.51	1.0	0.675	0.675	0.0
bigrams	0.57	0.51	11.8	0.582	0.551	0.566	0.675	-16.1
GBGs	0.572	0.51	12.2	0.574	0.621	0.597	0.675	-11.6
OSBs	0.565	0.51	10.8	0.565	0.637	0.599	0.675	-11.3

Table 4.34: Support vector machine over tiles, trained on one of three transcript types **(Waco not included)**

The SVM results are shown in Table 4.34. The Taylor experiments outperformed the Rogan experiments. This was due to similar levels of precision, but increased recall in the Taylor set. The experiments trained without using the San Diego Police transcript generally resulted in the highest F-scores, but this is misleading. These high F-scores were achieved by calling almost all of the test set persuasive.

4.15.4 Effects of Elimimating Waco and Single Transcript Type Training for Tiles

In general, each set of experiments produced increases in precision, recall and F-score, when compared to the results in section 4.9. The improved scores were more significant for the Taylor experiments than for the Rogan experiments. However, these results should not be used as the basis for any conclusions due to the poor F-scores and the trends shown in previous experiments over tiles.

4.16 Majority Voting over Posts

Having explored the extent to which individual machine learning techniques could be used to detect persuasion, this research investigated the utility of voting schemes using these techniques collectively. Two separate voting schemes were described in section 3.9. The results of these experiments over posts appear in this section.

4.16.1 Six-fold Validation

Repetition 1								
Features	Accuracy	Baseline Accuracy	% Change	Precision	Recall	F-Score	Baseline F-score	% Change
unigrams	0.896	0.887	1.0	0.548	0.375	0.442	0.202	118.8
bigrams	0.899	0.887	1.4	0.584	0.32	0.411	0.202	103.5
GBGs	0.897	0.887	1.1	0.57	0.314	0.402	0.202	99.0
OSBs	0.9	0.887	1.5	0.61	0.277	0.379	0.202	87.6
Repetition 2								
Features	Accuracy	Baseline Accuracy	% Change	Precision	Recall	F-Score	Baseline F-score	% Change
unigrams	0.895	0.886	1.0	0.554	0.374	0.444	0.203	118.7
bigrams	0.898	0.886	1.4	0.587	0.333	0.42	0.203	106.9
GBGs	0.895	0.886	1.0	0.563	0.315	0.402	0.203	98.0
OSBs	0.899	0.886	1.5	0.615	0.271	0.374	0.203	84.2
Repetition 3								
Features	Accuracy	Baseline Accuracy	% Change	Precision	Recall	F-Score	Baseline F-score	% Change
unigrams	0.897	0.888	1.0	0.546	0.361	0.431	0.201	114.4
bigrams	0.9	0.888	1.4	0.59	0.333	0.419	0.201	108.5
GBGs	0.899	0.888	1.2	0.579	0.331	0.414	0.201	106.0
OSBs	0.899	0.888	1.2	0.598	0.28	0.374	0.201	86.1
Repetition 4								
Features	Accuracy	Baseline Accuracy	% Change	Precision	Recall	F-Score	Baseline F-score	% Change
unigrams	0.898	0.889	1.0	0.55	0.357	0.432	0.199	117.1
bigrams	0.898	0.889	1.0	0.565	0.303	0.392	0.199	97.0
GBGs	0.898	0.889	1.0	0.565	0.309	0.396	0.199	99.0
OSBs	0.899	0.889	1.1	0.598	0.265	0.363	0.199	82.4
Repetition 5								
Features	Accuracy	Baseline Accuracy	% Change	Precision	Recall	F-Score	Baseline F-score	% Change
unigrams	0.896	0.886	1.1	0.56	0.366	0.442	0.204	116.7
bigrams	0.899	0.886	1.5	0.595	0.321	0.415	0.204	103.4
GBGs	0.9	0.886	1.6	0.609	0.331	0.429	0.204	110.3
OSBs	0.9	0.886	1.6	0.634	0.281	0.388	0.204	90.2

Table 4.35: Majority voting over posts

Table 4.35 shows the results of using the three machine learning techniques to vote for classification. Any post that received two or more votes for "persuasion" was classified as persuasive. There was no significant change in F-score (less than .01) compared to maximum entropy experiments using unigrams (see Table 4.5). Voting experiments for bigrams had changes in F-score greater than than .01. These improvements in F-score resulted from a universal increase in recall

across all three feature sets. This increase in recall was accompanied by a increase in precision for gappy bigrams and OSBs. Bigram voting experiments exhibited a decrease in precision.

In comparison to the naive bayes results in Table 4.6, only one voting experiment improved in F-score by .01 or more (repetition 3, bigrams). Three unigram voting experiments had less than a .01 change in F-score (repetitions 1, 2 and 4). All other voting experiments resulted in F-score decreases of .01 or greater. For all experiments, precision was higher, and recall was lower.

The voting experiments showed a .01 or more increase in F-score for unigrams when compared to the SVM results in Table 4.7. There was also an increase in F-score of .01 or more for three of the five experiments using gappy bigrams (repetitions 3, 4, and 5). The increases in F-score resulted from increases in both recall and precision. All other voting experiments had F-score changes less than .01. All voting experiments were more precise than the SVM only experiments. Recall was higher for unigram and gappy bigram voting experiments, but lower for bigrams and OSBs.

This set of results suggest that a majority voting scheme is an improvement over a maximum entropy only classifier and a slight improvement over and SVM only classifier. This voting scheme does not perform as well as a naive bayes only classifier.

4.16.2 Leave-One-Out

Table 4.36 shows that there was no significant change in F-score (less than .01) compared to maximum entropy experiments using unigrams (see Table 4.11). Leaving out single transcript types during training for bigrams resulted in F-score changes greater than than .01 across all leave. These improvements in F-score resulted from an increase in recall with a decrease in precision. Both the Taylor and the San Diego Police experiments had F-score increases of .01 or more for gappy bigrams, while the Rogan and the Waco F-scores showed no change. The Rogan, Taylor, and San Diego Police experiments had F-score increases of .01 or more, while the Waco F-score showed no change.

In comparison to the naive bayes results in Table 4.12, the results of the Rogan voting experiments showed F-score decreased of .01 or more for all feature sets. The Taylor experiments resulted in F-score decreases of .01 or more for all three types of bigrams with an F-score increase for unigrams. Using voting, the San Diego Police experiments resulted in F-score increases of .01 or more for unigrams, bigram, and OSBs. Gappy bigrams resulted in a decrease of 3%. Voting produced a decrease in F-score for all Waco experiments, except unigrams.

Trained with all, except Rogan (13608 posts, 89.7% not persuasive)								
Features	Accuracy	Baseline Accuracy	% Change	Precision	Recall	F-Score	Baseline F-score	% Change
unigrams	0.866	0.85	1.9	0.598	0.329	0.425	0.262	62.2
bigrams	0.875	0.85	2.9	0.693	0.304	0.422	0.262	61.1
GBGs	0.875	0.85	2.9	0.691	0.304	0.422	0.262	61.1
OSBs	0.873	0.85	2.7	0.731	0.252	0.374	0.262	42.7
Trained with all, except Taylor (11944 posts, 89.7% not persuasive)								
Features	Accuracy	Baseline Accuracy	% Change	Precision	Recall	F-Score	Baseline F-score	% Change
unigrams	0.877	0.872	0.6	0.526	0.367	0.432	0.227	90.3
bigrams	0.883	0.872	1.3	0.601	0.267	0.369	0.227	62.6
GBGs	0.882	0.872	1.1	0.588	0.272	0.372	0.227	63.9
OSBs	0.884	0.872	1.4	0.642	0.215	0.322	0.227	41.9
Trained with all, except SDPolice (18033 posts, 88.7% not persuasive)								
Features	Accuracy	Baseline Accuracy	% Change	Precision	Recall	F-Score	Baseline F-score	% Change
unigrams	0.899	0.826	8.8	0.811	0.546	0.653	0.297	119.9
bigrams	0.911	0.826	10.3	0.897	0.553	0.684	0.297	130.3
GBGs	0.896	0.826	8.5	0.835	0.504	0.628	0.297	111.4
OSBs	0.9	0.826	9.0	0.895	0.482	0.627	0.297	111.1
Trained with all, except Waco (12986 post, 86.0% not persuasive)								
Features	Accuracy	Baseline Accuracy	% Change	Precision	Recall	F-Score	Baseline F-score	% Change
unigrams	0.909	0.937	-3.0	0.288	0.293	0.291	0.119	144.5
bigrams	0.914	0.937	-2.5	0.328	0.347	0.337	0.119	183.2
GBGs	0.906	0.937	-3.3	0.289	0.333	0.31	0.119	160.5
OSBs	0.914	0.937	-2.5	0.309	0.285	0.297	0.119	149.6

Table 4.36: Majority voting over posts, trained on three of four transcript types

The voting experiments showed a .01 or more increase in F-score for unigrams when compared to the SVM results in Table 4.13, with the exception of the Taylor experiments, which showed no change. The Rogan experiments resulted in no change when using bigrams and gappy bigrams with an F-score decrease of more than .02 for OSBs. Bigrams resulted in an F-score change greater than .01 for the Taylor experiments. The Waco experiments produced F-score increases of more than 2% for bigrams and gappy bigrams, with no change for OSBs.

This set of results reinforces that a majority voting scheme is an improvement over a maximum entropy only classifier and that the Waco transcripts are different than the other transcripts.

4.16.3 Leave-One-In

Table 4.37 shows that there was an increase in F-score (greater than .01) compared to maximum entropy experiments using all three types of bigrams (see Table 4.17). All four experiments showed no change when using unigrams. When using only the Waco transcripts as training data, the voting experiments showed no change for bigrams and OSBs and more than a .025 decrease in F-score for gappy bigrams.

Trained with only Rogan (5249 posts, 85.0% not persuasive)								
Features	Accuracy	Baseline Accuracy	% Change	Precision	Recall	F-Score	Baseline F-score	% Change
unigrams	0.889	0.897	-0.9	0.455	0.414	0.434	0.187	132.1
bigrams	0.895	0.897	-0.2	0.481	0.293	0.364	0.187	94.7
GBGs	0.896	0.897	-0.1	0.489	0.336	0.398	0.187	112.8
OSBs	0.901	0.897	0.4	0.533	0.274	0.362	0.187	93.6
Trained with only Taylor (6913 posts, 87.2% not persuasive)								
Features	Accuracy	Baseline Accuracy	% Change	Precision	Recall	F-Score	Baseline F-score	% Change
unigrams	0.887	0.891	-0.4	0.477	0.358	0.409	0.197	107.6
bigrams	0.897	0.891	0.7	0.547	0.308	0.394	0.197	100.0
GBGs	0.896	0.891	0.6	0.543	0.324	0.406	0.197	106.1
OSBs	0.899	0.891	0.9	0.576	0.271	0.368	0.197	86.8
Trained with only SDPolice (824 posts, 82.9% not persuasive)								
Features	Accuracy	Baseline Accuracy	% Change	Precision	Recall	F-Score	Baseline F-score	% Change
unigrams	0.868	0.887	-2.1	0.41	0.385	0.397	0.204	94.6
bigrams	0.889	0.887	0.2	0.531	0.184	0.273	0.204	33.8
GBGs	0.889	0.887	0.2	0.521	0.233	0.322	0.204	57.8
OSBs	0.891	0.887	0.5	0.566	0.175	0.268	0.204	31.4
Trained with only Waco (5871 posts, 93.7% not persuasive)								
Features	Accuracy	Baseline Accuracy	% Change	Precision	Recall	F-Score	Baseline F-score	% Change
unigrams	0.866	0.86	0.7	0.56	0.194	0.288	0.246	17.1
bigrams	0.867	0.86	0.8	0.734	0.082	0.148	0.246	-39.8
GBGs	0.867	0.86	0.8	0.69	0.087	0.155	0.246	-37.0
OSBs	0.865	0.86	0.6	0.787	0.053	0.099	0.246	-59.8

Table 4.37: Majority voting over posts, trained on one of four transcript types

In comparison to the naive bayes results in Table 4.18, the results of all voting experiments showed F-score decreased of .01 or more for OSBs. The Rogan, Taylor and San Diego Police experiments resulted in F-score increases of .01 or more for unigrams and decreases of .01 or more for gappy bigrams. The Waco experiments resulted in the opposite F-score trend, a decrease for unigrams and an increase gappy bigrams. Bigrams resulted in F-score decreased for the Rogan and San Diego Police experiments. The Taylor experiment produced an F-score increase for bigrams, while the Waco experiment produced no change.

The Rogan voting experiments showed an increase in F-score of more than .01 for unigrams, with no significant changes for all other feature sets, when compared to the SVM results in Table 4.19. The Taylor experiments resulted in a more than .01 increase in F-score for unigrams and a more than .01 decrease for OSBs. The results for bigrams and gappy bigrams did not change significantly. Using bigrams and OSBs for the San Diego Police experiments resulted in a more than .01 decrease in F-score. Unigrams produced an F-score increase of more than .01, while gappy bigrams resulted in no significant change. The Waco F-score results were all more than .01 below the SVM results in Table 4.19.

The application of majority voting scheme in this set of experiment shows an improvement over a maximum entropy only classifier and that the Waco transcripts are sufficiently different from the rest of the corpus.

4.16.4 Leave-One-Out without Waco

Trained with all, except Rogan (7737 posts, 86.7% not persuasive)								
Features	Accuracy	Baseline Accuracy	% Change	Precision	Recall	F-Score	Baseline F-score	% Change
unigrams	0.861	0.85	1.3	0.56	0.358	0.437	0.262	66.8
bigrams	0.871	0.85	2.5	0.644	0.315	0.423	0.262	61.5
GBGs	0.871	0.85	2.5	0.636	0.333	0.437	0.262	66.8
OSBs	0.875	0.85	2.9	0.699	0.292	0.412	0.262	57.3
Trained with all, except Taylor (6073 posts, 84.7% not persuasive)								
Features	Accuracy	Baseline Accuracy	% Change	Precision	Recall	F-Score	Baseline F-score	% Change
unigrams	0.874	0.872	0.2	0.508	0.399	0.447	0.227	96.9
bigrams	0.881	0.872	1.0	0.565	0.303	0.394	0.227	73.6
GBGs	0.883	0.872	1.3	0.571	0.344	0.429	0.227	89.0
OSBs	0.884	0.872	1.4	0.605	0.276	0.379	0.227	67.0
Trained with all, except SDPolice (12162 posts, 86.3% not persuasive)								
Features	Accuracy	Baseline Accuracy	% Change	Precision	Recall	F-Score	Baseline F-score	% Change
unigrams	0.9	0.826	9.0	0.788	0.582	0.669	0.297	125.3
bigrams	0.91	0.826	10.2	0.84	0.596	0.697	0.297	134.7
GBGs	0.909	0.826	10.0	0.832	0.596	0.694	0.297	133.7
OSBs	0.902	0.826	9.2	0.86	0.525	0.652	0.297	119.5

Table 4.38: Majority voting over posts, trained on two of three transcript types **(Waco not included)**

Table 4.38 shows that there was an increase in F-score compared to maximum entropy experiments using all three types of bigrams (see Table 4.23). All increases were greater than .01, with the exception of the Rogan gappy bigram experiment (.009 change). The San Diego Police experiment showed no change in F-score when using unigrams. Both the Taylor and Rogan experiments showed a decrease in F-score of more than .01.

In comparison to the naive bayes results in Table 4.24, there was a decrease in F-score for all gappy bigram and OSB experiments. Both the Rogan and Taylor experiments resulted in an increase in F-score, while the San Diego Police experiments resulted in a decrease in F-score. The San Diego Police experiment posted a increase in F-score for bigrams, while both the Rogan and the Taylor experiments posted a decrease in F-score.

The Rogan voting experiments showed an increase in F-score of more than .01 for unigrams with a decrease in F-score for both bigrams and OSBs, when compared to the SVM results in Table 4.25. Gappy bigrams resulted in no significant change in F-score. The Taylor experiments resulted in a more than .01 decrease in F-score for bigrams, and no change for all other feature

sets. Using unigrams and OSBs for the San Diego Police experiments resulted in a more than .01 increase in F-score. Bigrams produced no change in F-score, while gappy bigrams resulted in a greater .01 decrease in F-score.

This set of experiments shows an even greater improvement than the experiments in section 4.12.

4.16.5 Leave-One-In without Waco

Trained with only Rogan (5249 posts, 85.0% not persuasive)								
Features	Accuracy	Baseline Accuracy	% Change	Precision	Recall	F-Score	Baseline F-score	% Change
unigrams	0.875	0.867	0.9	0.537	0.434	0.48	0.235	104.3
bigrams	0.878	0.867	1.3	0.58	0.304	0.399	0.235	69.8
GBGs	0.882	0.867	1.7	0.599	0.343	0.436	0.235	85.5
OSBs	0.885	0.867	2.1	0.654	0.291	0.403	0.235	71.5
Trained with only Taylor (6913 posts, 87.2% not persuasive)								
Features	Accuracy	Baseline Accuracy	% Change	Precision	Recall	F-Score	Baseline F-score	% Change
unigrams	0.867	0.846	2.5	0.604	0.389	0.473	0.266	77.8
bigrams	0.876	0.846	3.5	0.707	0.325	0.446	0.266	67.7
GBGs	0.877	0.846	3.7	0.705	0.346	0.464	0.266	74.4
OSBs	0.877	0.846	3.7	0.754	0.294	0.423	0.266	59.0
Trained with only SDPolice (824 posts, 82.9% not persuasive)								
Features	Accuracy	Baseline Accuracy	% Change	Precision	Recall	F-Score	Baseline F-score	% Change
unigrams	0.855	0.862	-0.8	0.47	0.394	0.428	0.242	76.9
bigrams	0.873	0.862	1.3	0.64	0.184	0.286	0.242	18.2
GBGs	0.875	0.862	1.5	0.619	0.234	0.34	0.242	40.5
OSBs	0.876	0.862	1.6	0.69	0.18	0.286	0.242	18.2

Table 4.39: Majority voting over posts, trained on one of three transcript types **(Waco not included)**

Table 4.39 shows that there was an increase in F-score compared to maximum entropy experiments using all three types of bigrams (see Table 4.29). The San Diego Police experiment showed no change in F-score when using unigrams. Both the Taylor and Rogan experiments showed a decrease in F-score of more than .01.

In comparison to the naive bayes results in Table 4.30, there was a decrease in F-score for all three types of bigram experiments, with one exception. The Taylor experiment using bigrams resulted in no significant change in F-score. When using unigrams as feature, both the Rogan and Taylor experiments resulted in an increase in F-score, while the San Diego Police experiments showed no change in F-score.

All voting experiments showed an increase in F-score of more than .01 for unigrams with a decrease in F-score for both bigrams and OSBs, when compared to the SVM results in Table 4.31. Gappy bigrams resulted in no significant change in F-score.

4.17 Single Classifier Voting over Posts

The results in section 4.16 showed that a majority voting scheme could produced better results than maximum entropy only and SVM only classifiers. A single classifier voting scheme was presented in section 3.9. This voting scheme was briefly explored, and a selection of the results of these experiments over posts appear in this section.

Trained will all, except Rogan (7737 posts, 86.7% not persuasive)								
Features	Accuracy	Baseline Accuracy	% Change	Precision	Recall	F-Score	Baseline F-score	% Change
unigrams	0.841	0.85	-1.1	0.473	0.51	0.491	0.262	87.4
bigrams	0.835	0.85	-1.8	0.459	0.546	0.499	0.262	90.5
GBGs	0.856	0.85	0.7	0.517	0.593	0.553	0.262	111.1
OSBs	0.859	0.85	1.1	0.53	0.526	0.528	0.262	101.5
Trained will all, except Taylor (6073 posts, 84.7% not persuasive)								
Features	Accuracy	Baseline Accuracy	% Change	Precision	Recall	F-Score	Baseline F-score	% Change
unigrams	0.847	0.872	-2.9	0.425	0.551	0.48	0.227	111.5
bigrams	0.833	0.872	-4.5	0.396	0.58	0.471	0.227	107.5
GBGs	0.842	0.872	-3.4	0.42	0.611	0.498	0.227	119.4
OSBs	0.846	0.872	-3.0	0.425	0.579	0.49	0.227	115.9
Trained will all, except SDPolice (12162 posts, 86.3% not persuasive)								
Features	Accuracy	Baseline Accuracy	% Change	Precision	Recall	F-Score	Baseline F-score	% Change
unigrams	0.9	0.826	9.0	0.711	0.716	0.714	0.297	140.4
bigrams	0.895	0.826	8.4	0.684	0.738	0.71	0.297	139.1
GBGs	0.904	0.826	9.4	0.698	0.787	0.74	0.297	149.2
OSBs	0.899	0.826	8.8	0.719	0.688	0.703	0.297	136.7

Table 4.40: Single Classifier Voting Over Posts,trained on two of three transcript types **(Waco not included)**

When the results in Table 4.40 are compared to the results in Table 4.38, the most significant trend is an increase in recall for all experiments. This is an important change due to the fact that all of the F-scores present in this chapter are primarily driven by recall. Since all three classifiers exhibited higher precision that recall for post experiments, this voting scheme classifies the maximum number of posts as persuasive with only a slight decrease in percision. Voting schemes similar to this should be the focus of future work.

4.18 Conclusion

The results presented in this chapter highlight important information about the feature sets and the machine learning techniques. The most important conclusion on these results is that machine learning techniques were able to detect to persuasion, albeit to a limited degree. While several experiments outperformed the baseline F-score by over 120%, there is substantial room for improvement.

The results of the 6-fold validation experiments showed that naive bayes performed well using gappy bigrams over posts, while maximum entropy and SVM performed better with unigrams. Contrary to our initial hypothesis, OSBs did not perform significantly better than any of the other feature sets. However, they often had the highest precision, which indicates that they may be of use in future reseach as part of a larger, more diverse feature set.

All three methods exhibited low recall and high precision for both types of segmentation. However, the results presented in this chapter did not give any evidence that TextTiling was a useful segmentation scheme for these classification tasks. However, this may be primarily attributed to labeling scheme applied to the tiles (see section 3.4).

Our results also revealed some important information about the transcripts in the corpus. We learned that the Waco transcripts are substantially different from the other three types of transcripts. Our results suggest that Taylor and Rogan are similar on the post level. Additionally, it seems that the models learned from using the Taylor transcripts as training data generalize more easily than the model learned from the Rogan transcripts. Having explored previous and related work, detailed our experimental design, and presented our results, the last step is to make some final conclusions and to suggest future work for research in this field.

CHAPTER 5:
Conclusion

5.1 Summary

This thesis addressed the question, "Can we learn to identify persuasion as as characterized by Cialdini's model using traditional machine learning techniques?" As shown below, we answer this question with a weak "yes."

In Chapter 2, we presented relevant concepts and previous work that were useful for this research. We described in detail Cialdini's persuasion model and how it could be applied to machine learning. We described and identified feature sets that were later used for our experiments. We presented a suite of metrics needed to evaluate our hypotheses. Finally, we concluded the chapter with a presentation of software tools that enabled this research.

Having presented all the relevant concepts, features, metrics, and tools, the next task was to elaborate on our experimental design. In Chapter 3, we presented a description of the data that was created at the Naval Postgraduate School in the Natural Language Processing Lab. We detailed the process associated with making the data usable for machine learning, providing a step-by-step framework for future research using this data. Additionally, we outlined the details of the experimental setup for each machine learning technique.

Next we reviewed and analyzed the results of our experiments in Chapter 4. Our results revealed important trends and characteristics about the behavior of our feature sets and machine learning techniques. The results showed that naive bayes performed well using gappy bigrams over posts, while maximum entropy and SVM performed better with unigrams. This may be due to the fact that maximum entropy and support vector machines are discriminative approaches, while naive bayes is a generative approach. Discriminative models indicate which class is most similar. For an inter-word distance of 5, OSBs and gappy bigrams may introduce information that distorts this similarity. The results for tiles were not as encouraging. This suggests that TextTiling was not a useful segmentation scheme for these classification tasks. This may be due to the labeling scheme described in section sectTiling.

Our results also revealed that the Waco transcripts are substantially different from the other three types of transcripts and that the Taylor transcripts and Rogan transcripts are similar on the

post level. Additionally, the models learned from using the Taylor transcripts as training data seemed to generalize more easily than models learned from the Rogan transcripts. While most of our experiments over posts showed an improvement over the baseline F-scores, none of our experiments resulted in high F-scores that would indicate that this problem has been solved. The next section provided insight into what future work would be needed to solve this classification problem.

5.2 Future Work

5.2.1 Data Set Improvements

After conducting this research, it is clear that there is a need for more and larger data sets annotated for belief. NPS has the only data set of this kind. The results presented in Chapter 4, showed that one of the four types of transcripts in this corpus is significantly different from the others. Additionally, this data set is limited to negotiator transcripts. For any real world applications of persuasion detection, especially DoD and intelligence applications, the item of interest will not be a negotiation transcript. The items of interest will be Web pages, blogs, SMS messages, speech recordings, and other media. These data sets may prove more useful as they will include only features produced by the writers or the speakers.

Since this data set already exists, the next step should be to improve and to expand it. Improvements could include other information about particular posts, such as distance from the previous persuasive post, correct speaker tags and dialogue act tags, as well as adding more negotiation transcripts to the corpus.

5.2.2 Feature Set Improvements

This research used only features that are artifacts of the words spoken by the actual negotiation participants. Using gappy bigrams and OSBs did help in some cases. Future research should explore the effect on the maximum distance between words and its effect on accuracy and recall. The experiments in this research only used one type of feature to form the feature sets. It is possible that combining higher recall features, such as unigrams, and high precision features such as OSBs may obtain better results.

Future work with improved data sets may include more features such as character bigram and trigrams. However, there are other directions to explore for features. One approach could be to build topic models for persuasion. These topics models could then be used in conjunction with the machine learning techniques used in this research, as well as other machine learning

techniques. If data sets for SMS and blog existed, it may be possible to use other features, such as the number of recipients or number of comments posted.

5.2.3 Future Research

There are many areas to pursue for future work. Each machine learning technique presented has parameters that can be tuned. While the method presented in this research is a reasonable first attempt, there are more sophisticated methods available. This research has also questioned the usefulness of texttiling for dialogues. It is possible that tiles could be useful given a better labeling scheme, such as using the most common post label within the tile. Another approach might separate the two sides of the negotiation and then apply TextTiling in conjunction with an improved labeling scheme.

Improved persuasion detection on the post level may prove difficult because posts are often short and their length in the corpus varies considerably, similar to other tasks using chat messages and SMS. Methods for better segmentation or normalization for length should be explored and applied to this data set. Another possibility is to conduct research investigating human subject agreement on tagging the beginning and the end of a persuasive section of a conversation. These sections could then be used for machine learning experiments.

Since negotiations are a series of turns, future work should included using Markov chain model, which take sequence and time into account. At the beginning of a negotiation, the hostage taker usually tries to secure as many demands as possible. The responding agency negotiator typically does not give into these demands at the outset. As time passes, a hostage taker may be willing to settle for a subset of his demands, where as a hostage negotiator will ask for larger and larger concessions. One of the more common types of persuasion in a hostage negotiation is commitment and consistency, which requires a offer-accept-enforce sequence. These characteristics all reinforce that time and sequence play a part in the presence of persuasive.

While this research explored voting as a possible means to improve the results of the experiments, further research should explore the utility of bagging and boosting. This exploration should include some of the weak classifiers in this thesis, as well as new classifiers that may result from future work in this area. In this research, F-score was affected primarily by low recall. If it is possible to find classifiers that have high recall, but do not result to labeling the entire test set as one class, these could be cascaded or chained with some of the higher precision classifiers in this research. In addition, this research only explored voting within a feature set.

Other possibilities could include expanded voting schemes that use the same machine learning method but combine the votes across features or that use different machine learning methods but combine the votes across features.

The experiments described in Chapter 3 did not take speaker into account when making predictions. It may be possible to build generative models that account for speaker when making a prediction. Since both sides of a negotiation are talking about the same topic, it is likely that they will use the same set of words or the same types of words. However, each side has its own goal. Consequently, it might be the case that features that are indicative of persuasion for one side of the conversation, may not be for the other.

This research did not use not account for parts of speech or syntax. It may be the case that the part of speech or syntax structure for persuasion is distinctly different from non-persuasive portions of a conversation. If this is the case, then there will be different parts of speech and parse trees that will indicated the class. Modals, such as "should" and "ought," could be one part of speech that may have an important role in persuasion detection. Once the binary classification task has been solved, the next task will be to identify a particular type of persuasion.

5.3 Concluding Remarks

Persuasion detection has proved to be a difficult problem. Within the Department of Defense and the intelligence community, there is a need to know when persuasion is happening. There are two scenarios where this is important. One is where our enemies are trying to influence friendly or neutral parties to act against the United States. In this scenario, persuasion detection enables us know that someone means do us harm and the audience that they are trying target. With this information, we now would have the ability to respond in an appropriate and timely manner. The second scenario where this is important is in learning the persuasion model of another culture. If we could detect persuasion in other languages, then we could learn how the dominant group is influencing the local populace. The lessons that we have learned in Iraq and Afghanistan, have taught us that culture is important. In both of these wars, the population is the center of gravity for both sides. The populace is often targeted with radio, television, and print media. With a model of their persuasion techniques, we would now be able to conduct effective and targeted information and psychological operations to sway the populace in support of our forces. Given these two scenarios, persuasion detection should continue to be a focus of DoD and intelligence research.

REFERENCES

[1] Joint Chiefs of Staff, *Joint Vision 2020*. Washington D.C.: U.S. Goverment Printing Office, 2000.

[2] W.-H. Lin, T. Wilson, J. Wiebe, and A. Hauptmann, "Which side are you on?: identifying perspectives at the document and sentence levels," in *CoNLL-X '06: Proceedings of the Tenth Conference on Computational Natural Language Learning*. Morristown, NJ, USA: Association for Computational Linguistics, 2006, pp. 109–116.

[3] D. Bikel and J. Sorensen, "If we want your opinion," in *ICSC '07: Proceedings of the International Conference on Semantic Computing*. Washington D.C.: IEEE Computer Society, 2007, pp. 493–500.

[4] H. T. Gilbert, "Persuasion detection in conversation," Master's thesis, Naval Postgraduate School, Monterey, CA, 2010.

[5] R. Cialdini, *Influence: The Psychology of Persuasion*. New York, NY: Collins, 2007.

[6] T. Joachims, "A probabilistic analysis of the rocchio algorithm with tfidf for text categorization," in *ICML '97: Proceedings of the Fourteenth International Conference on Machine Learning*. San Francisco, CA: Morgan Kaufmann Publishers Inc., 1997, pp. 143–151.

[7] H. Lodhi, C. Saunders, J. Shawe-Taylor, N. Cristianini, and C. Watkins, "Text classification using string kernels," *The Journal of Machine Learning Research*, vol. 2, p. 444, 2002.

[8] N. Cancedda, E. Gaussier, C. Goutte, and J. Renders, "Word sequence kernels," *The Journal of Machine Learning Research*, vol. 3, pp. 1059–1082, 2003.

[9] G. V. Cormack, J. M. Gómez Hidalgo, and E. P. Sánz, "Spam filtering for short messages," in *CIKM '07: Proceedings of the Sixteenth ACM Conference on Information and Knowledge Management*. Lisbon, Portugal: ACM, 2007, pp. 313–320.

[10] W. Yerazunis, "Sparse binary polynomial hashing and the CRM114 discriminator," in *2003 Spam Conference*. Cambridge, MA: MIT, 2003.

[11] C. Siefkes, F. Assis, S. Chhabra, and W. Yerazunis, "Combining winnow and orthogonal sparse bigrams for incremental spam filtering," *Knowledge Discovery in Databases*, pp. 410–421, 2004.

[12] C. Fox, "A stop list for general text," *SIGIR Forum*, vol. 24, no. 1-2, pp. 19–21, 1989.

[13] A. McCallum and K. Nigam, "A comparison of event models for naive bayes text classification," *AAAI-98 Workshop on Learning for Text Categorization*, vol. 752, 1998.

[14] M. Hearst, "Texttiling: segmenting text into multi-paragraph subtopic passages," *Computational Linguistics*, vol. 23, no. 1, pp. 33–64, 1997.

[15] T. Nomoto and Y. Nitta, "A grammatico-statistical approach to discourse partitioning," in *Proceedings of the 15th Conference on Computational Linguistics*. Morristown, NJ: Association for Computational Linguistics, 1994, pp. 1145–1150.

[16] D. Jurafsky, J. Martin, and A. Kehler, *Speech and Language Processing: An introduction to natural language processing, computational linguistics, and speech recognition*. Cambridge, MA: MIT Press, 2000.

[17] A. Berger, V. Pietra, and S. Pietra, "A maximum entropy approach to natural language processing," *Computational Linguistics*, vol. 22, no. 1, pp. 39–71, 1996.

[18] A. Ratnaparkhi, "Maximum entropy models for natural language ambiguity resolution," Ph.D. dissertation, University of Pennsylvania, Philadelphia, PA, 1998.

[19] S. Della Pietra, V. Della Pietra, J. Lafferty, R. Technol, and S. Brook, "Inducing features of random fields," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 19, no. 4, pp. 380–393, 1997.

[20] H. Daumé III, "Notes on CG and LM-BFGS optimization of logistic regression," August 2004, paper available at http://pub.hal3.name#daume04cg-bfgs, implementation available at http://hal3.name/megam/ [Accessed: May 29, 2010].

[21] (2008) Support vector machine — Wikipedia, the free encyclopedia. [Online]. Available: http://en.wikipedia.org/wiki/Support_vector_machine [Accessed: April 13, 2010].

[22] E. Alpaydin, *Introduction to Machine Learning*. Cambridge, MA: MIT Press, 2004.

[23] C. Cortes and V. Vapnik, "Support-vector networks," *Machine Learning*, vol. 20, no. 3, pp. 273–297, 1995.

[24] C. Chang and C. Lin. (2001) LIBSVM: A library for support vector machines. [Online]. Available: http://www.csie.ntu.edu.tw/~cjlin/libsvm [Accessed: June 10, 2010].

[25] C. Hsu, C. Chang, C. Lin *et al.* (2003) A practical guide to support vector classification. [Online]. Available: http://www.csie.ntu.edu.tw/~cjlin/papers/guide/guide.pdf [Accessed: June 10, 2010].

[26] J. Lewis, "A short SVM (support vector machine) tutorial," *CGIT Lab/IMSC, Univerisity of Southern California*, 2004.

[27] S. Gunn, "Support vector machines for classification and regression," University of Southampton, Southampton, England, Tech. Rep., 1998.

[28] M. Hearst, S. Dumais, E. Osman, J. Platt, and B. Scholkopf, "Support vector machines," *IEEE Intelligent Systems and Their Applications*, vol. 13, no. 4, pp. 18–28, 1998.

[29] C. Van Rijsbergen. (1979) Information retrieval. London, England. [Online]. Available: http://www.dcs.gla.ac.uk/Keith/Preface.html [Accessed: June 10, 2010].

[30] F. Provost, T. Fawcett, and R. Kohavi, "The case against accuracy estimation for comparing classifiers," in *Proceedings of the Fifteenth International Conference on Machine Learning*, 1998, pp. 445–453.

[31] K. Ivanov, "Quality-Control of Information: On the concept of accuracy of information in data-banks and in management information systems," Ph.D. dissertation, Royal Institute of Technology, Stockholm, Sweden, 1972.

[32] A. Schein. (2010) The Naval Postgraduate School Machine Learning Library. Monterey, CA. [Online]. Available: http://sourceforge.net/projects/npsml/ [Accessed: May 29, 2010].

[33] P. Baldi, P. Frasconi, and P. Smyth, *Modeling the Internet and the Web*. Chichester, England: Wiley, 2003.

[34] A. Ng and M. Jordan, "On Discriminative Vs. Generative Classifiers: A comparison of logistic regression and naive bayes," *Advances in Neural Information Processing Systems*, vol. 2, pp. 841–848, 2002.

THIS PAGE INTENTIONALLY LEFT BLANK

Initial Distribution List

1. Defense Technical Information Center
   Fort Belvoir, Virginia

2. Dudley Knox Library
   Naval Postgraduate School
   Monterey, California

3. Marine Corps Representative
   Naval Postgraduate School
   Monterey, California

4. Director, Training and Education, MCCDC, Code C46
   Quantico, Virginia

5. Director, Marine Corps Research Center, MCCDC, Code C40RC
   Quantico, Virginia

6. Marine Corps Tactical Systems Support Activity (Attn: Operations Officer)
   Camp Pendleton, California